03-21-05

AF
2623
CD-ROM
IFW

Appl. No. : 09/488,390 Confirmation Number: 4399
Applicants : David M. Tumey, Tianning Xu
Filed : 01/19/2000
Title : Animated Toy Utilizing Artificial Intelligence and Facial Image Recognition
Assignee : Intelligent Verification Systems
TC/A.U. : 2623
Examiner : Vikkram Bali

Docket No. : Tumey.001

Honorable Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

37 CFR § 1.91 PETITION
TO MAKE EXHIBITS ACCOMPANYING 37 CFR §§ 1.131 and 1.132 AFFIDAVIT A PART OF THE OFFICIAL RECORD

Sir:

37 CFR § 1.91(a) provides that "[a] model or exhibit will not be admitted as part of the record of an application unless it: (1) substantially conforms to the requirements of § 1.52 or § 1.84; (2) is specifically required by the Office; or (3) is filed with a petition under this section including: (i) the fee set forth in § 1.17(h); and (ii) an explanation of why entry of the model or exhibit in the file record is necessary to demonstrate patentability."

Applicants have submitted an Affidavit of David M. Tumey under 37 CFR §§ 1.131 and 1.132 in an effort to swear behind a cited reference or alternatively to establish evidence of non-obviousness. Patent Office practice under 37 CFR § 1.131 requires Applicants to attach documentary evidence of their diligent efforts to reduce their invention to practice. Because much of the Applicants' evidence consists of <u>executable software modules</u> that have evidence of their "creation" dates, Applicants have submitted a CD containing those modules as part of Exhibit 8 to their Affidavit. Applicants seek to have this evidence entered into the record to

establish their diligent efforts to reduce their invention to practice. Furthermore, Applicants have submitted a CD as Exhibit 14 which demonstrates actual reduction to practice of the claimed invention. This second CD also establishes evidence of non-obviousness and the patentable worthiness of the invention. Applicants also wish to have these items entered as part of the permanent record for purposes of relying on them, if necessary, in an appeal.

Accordingly, Applicants respectfully petition the Commissioner to waive the requirements of 37 CFR § 1.91 as it pertains to the exhibits to Applicants' Affidavit.

As set forth in the accompanying fee transmittal form, the Commissioner, is authorized to deduct any fees that may be required from Gunn & Lee's deposit account no. 500808.

Respectfully submitted,

Eric Cernyar

Eric W. Cernyar
Reg. No. 45,919
Gunn & Lee, P.C.
700 N. St. Mary's St. Ste 1500
San Antonio, TX 78205
(210) 886-9500 (phone)
(210) 886-9883 (fax)
(210) 863-0063 (cell phone)

Appl. No.	:	09/488,390	Confirmation Number: 4399
Applicants	:	David M. Tumey, Tianning Xu	
Filed	:	01/19/2000	
Title	:	Animated Toy Utilizing Artificial Intelligence and Facial Image Recognition	
Assignee	:	Intelligent Verification Systems	
TC/A.U.	:	2623	
Examiner	:	Vikkram Bali	
Docket No.	:	Tumey.001	

Honorable Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

RESPONSE TO DECEMBER 8, 2004, OFFICE ACTION

Sir:

Remarks/Arguments begin on page 2 of this paper.

REMARKS:

Status of claims:

Claims 1-16 stand rejected as obvious over U.S. Patent No. 6,175,772 to Kamiya et al., for the same reasons set forth in the Examiner's January 30, 2004, and July 6, 2004, Office Actions. Applicants again respectfully traverse these rejections with both argument and affidavit evidence.

Applicants would note that they have twice amended the claims to make a distinction between mere "facial expression" recognition and "biometric" identification. In response to the January 30, 2004, Office Action, Applicants amended claims 1, 11, and 12 to add the following limitations:

Claim 1: "of a particular one of said animate or inanimate objects."
Claim 11: "of a particular person"
Claim 12: "indicative of a particular person."

In response to the July 6, 2004, Office Action, Applicants amended claims 1 and 12 even further to add the following limitations:

Claim 1: "operable to biometrically identify an imaged one of a plurality of animate or inanimate objects having facial or face-like characteristics by measuring the facial or face-like characteristics of the imaged object" and "biometric"
Claim 12: "biometrically"

U.S. Patent No. 6,175,772 to Kamiya et al does not disclose or teach biometric facial recognition. Nevertheless, the Examiner argues that "the claim limitations are given their broadest reasonable interpretations"[1] and maintains that Kamiya teaches "facial expression 'facial recognition'" as claimed.

Applicants respectfully urge that in light of the repeated amendments and arguments that Applicants have made regarding the claims, the claims cannot reasonably be given an interpretation that merely covers "facial expression" recognition without any capability of biometric identification.

Furthermore, the attached 37 §§ 1.131/1.132 Affidavit of David M. Tumey (with sworn verification by co-inventor Dr. Tony Xu) shows that biometric identification is not a trivial or obvious extension of mere face-expression recognition. Paragraph 24 notes that the ability to ascertain *emotive facial expressions* takes less processing than the ability to *uniquely identify a person*. In paragraph 25, Tumey and Dr. Xu attested that

[1] Applicants take this to mean that the Examiner construes claims 1-16 as reading on a toy that is capable only of recognizing "facial expressions."

"unique facial image recognition is significantly more complex than detecting a not-necessarily-unique facial expression."[2] Even the Kamiya reference makes this apparent – Kamiya's system only recognized seven basic emotional models – neutral, disgusted, happy, sad, surprised, angry, and fearful. Col. 6, lines 32-35; col. 7, lines 37-40.

One of the most significant challenges Tumey and Dr. Xu faced in reducing their invention to practice was processing delays. In paragraph 18 of their Affidavit, Tumey and Dr. Xu state that "[f]ace recognition algorithms that we had developed earlier for security applications were not fast enough to be used in a toy. They typically had about a 3-second delay between the image capture and the facial recognition – this delay was too long for a toy. Our goal was to make a toy that would recognize and respond to human users in real time." Also, Tumey and Dr. Xu's patent specification notes, on page 2, lines 8-9, that "identification processing delays can be excessive and unacceptable for many applications." And Tumey and Dr. Xu's survey of the prior art revealed that prior "approaches to providing an encoded facial image that could be stored, retrieved and compared, automatically or manually, at some later time for recognizing said human user" were not "viable for use in an articulated and animated toy or video game." Page 4, lines 20-24. Tumey and Dr. Xu's affidavit, attached, shows that they spent a great deal of time trying to develop efficient yet effective facial-image-recognition algorithms suitable for use with an interactive entertainment device.

Given these challenges and the limitations of the prior art, it cannot be maintained that the prior art provided any teaching, suggestion, or motivation (together with a reasonable expectation of success) to modify Kamiya, or combine it with other references, in order to read on the claims.

In any event, Tumey and Dr. Xu conceived of the claimed invention before the applicable section 102(e) date (April 13, 1998) of the Kamiya reference. *See* Affidavit ¶¶ 9-15. They also diligently reduced their invention to practice. *See* Affidavit ¶¶ 16-48. Applicants offer the attached 37 §§ 1.131/1.132 Affidavit of David M. Tumey (with sworn verification by co-inventor Dr. Tony Xu) in order to swear behind the Kamiya reference.

The undersigned is acutely aware of the Patent and Trademark Office's reluctance to allow patents in the business method/computer software/biometric identification/encryption field. Indeed, the PTO's own statistics show that allowances of patents in these fields have slowed down to a crawl. The undersigned presumes that much of that hostility derives not only from a spate of negative publicity, but also from the plethora of "concept patent applications" that have been filed in these technology

[2] *See also* Affidavit, at paragraph 28 ("Biometric identification, which again is completely different from feature/expression recognition, involves treating the entire face as a collection of unique features that are specific to each given individual, i.e., the distance between the eyes, nose and mouth coupled with the morphology of each feature ....").

areas. Many companies have filed applications with little more than the "germ" of an idea about a future technology without ever making any attempt to reduce their claimed invention to actual practice.

Tumey's and Dr. Xu's invention is not a mere "concept." With years of effort, they reduced their invention to practice. Attached as Exhibit 14 to their Affidavit is a copy of a CD with a video demonstration of a working prototype of a toy bear with facial image recognition abilities. This video is also available at http://www.cernyar.com/toy.mpg. Tumey's and Dr. Xu's efforts, and their willingness to share their inventive concepts with the world (through their application) are the kind of activities Congress intended the patent system to reward.

Applicants respectfully ask that the Examiner consider the attached Affidavit and Exhibits and withdraw the rejections. A petition under § 1.91 to make the exhibits a part of the record accompanies this response.

Conclusion

Believing that all things raised in the Examiner's December 8, 2004, Office Action have been addressed, the undersigned respectfully requests that the application be allowed and passed to issue.

As set forth in the accompanying petition for extension of time, the Commissioner, is authorized to deduct any additional fees (beyond the submitted check) that may be required from Gunn & Lee's deposit account no. 500808.

Respectfully submitted,

Eric Cernyar

Eric W. Cernyar
Reg. No. 45,919
Gunn & Lee, P.C.
700 N. St. Mary's St. Ste 1500
San Antonio, TX 78205
(210) 886-9500 (phone)
(210) 886-9883 (fax)
(210) 863-0063 (cell phone)

PTO/SB/17p (11-04)
Approved for use through 07/31/2007. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PETITION FEE **Under 37 CFR 1.17(f), (g) & (h)** **TRANSMITTAL** (Fees are subject to annual revision) **Send completed form to: Commissioner for Patents** **P.O. Box 1450, Alexandria, VA 22313-1450**		
	Application Number	09/488,390
	Filing Date	01/19/2000
	First Named Inventor	Tumey
	Art Unit	2623
	Examiner Name	Vikkram Bali
	Attorney Docket Number	Tumey.001

Enclosed is a petition filed under 37 CFR 1.91 that requires a processing fee (37 CFR 1.17(f), (g), or (h)). Payment of $ 130.00 is enclosed.
This form should be included with the above-mentioned petition and faxed or mailed to the Office using the appropriate Mail Stop (e.g., Mail Stop Petition), if applicable. *For transmittal of processing fees under 37 CFR 1.17(i), see form PTO/SB/17i.*

Payment of Fees (small entity amounts are NOT available for the petition fees)

☑ The Commissioner is hereby authorized to charge the following fees to Deposit Account No. 500808 :

☐ petition fee under 37 CFR 1.17(f), (g) or (h) ☑ any deficiency of fees and credit of any overpayments

Enclose a duplicative copy of this form for fee processing.

☑ Check in the amount of $ 190.00 is enclosed.

☐ Payment by credit card (Form PTO-2038 or equivalent enclosed). Do not provide credit card information on this form.

Petition Fees under 37 CFR 1.17(f): Fee $400 Fee Code 1462
For petitions filed under:
- **§ 1.53(e)** - to accord a filing date.
- **§ 1.57(a)** - to accord a filing date.
- **§ 1.182** - for decision on a question not specifically provided for.
- **§ 1.183** - to suspend the rules.
- **§ 1.378(e)** - for reconsideration of decision on petition refusing to accept delayed payment of maintenance fee in an expired patent.
- **§ 1.741(b)** - to accord a filing date to an application under § 1.740 for extension of a patent term.

Petition Fees under 37 CFR 1.17(g): Fee $200 Fee Code 1463
For petitions filed under:
- **§ 1.12** - for access to an assignment record.
- **§ 1.14** - for access to an application.
- **§ 1.47** - for filing by other than all the inventors or a person not the inventor.
- **§ 1.59** - for expungement of information.
- **§ 1.103(a)** - to suspend action in an application.
- **§ 1.136(b)** - for review of a request for extension of time when the provisions of section 1.136(a) are not available.
- **§ 1.295** - for review of refusal to publish a statutory invention registration.
- **§ 1.296** - to withdraw a request for publication of a statutory invention registration filed on or after the date the notice of intent to publish issued.
- **§ 1.377** - for review of decision refusing to accept and record payment of a maintenance fee filed prior to expiration of a patent.
- **§ 1.550(c)** - for patent owner requests for extension of time in ex parte reexamination proceedings.
- **§ 1.956** - for patent owner requests for extension of time in inter partes reexamination proceedings.
- **§ 5.12** - for expedited handling of a foreign filing license.
- **§ 5.15** - for changing the scope of a license.
- **§ 5.25** - for retroactive license.

Petition Fees under 37 CFR 1.17(h): Fee $130 Fee Code 1464
For petitions filed under:
- **§ 1.19(g)** - to request documents in a form other than that provided in this part.
- **§ 1.84** - for accepting color drawings or photographs.
- **§ 1.91** - for entry of a model or exhibit.
- **§ 1.102(d)** - to make an application special.
- **§ 1.138(c)** - to expressly abandon an application to avoid publication.
- **§ 1.313** - to withdraw an application from issue.
- **§ 1.314** - to defer issuance of a patent.

Michelle L Evans
Signature

03/18/05
Date

Michelle L. Evans
Typed or printed name

44,673
Registration No., if applicable

Appl. No.	:	09/488,390	Confirmation Number: 4399
Applicants	:	David M. Tumey, Tianning Xu	
Filed	:	01/19/2000	
Title	:	Animated Toy Utilizing Artificial Intelligence and Facial Image Recognition	
Assignee	:	Intelligent Verification Systems	
TC/A.U.	:	2623	
Examiner	:	Vikkram Bali	
Docket No.	:	Tumey.001	

AFFIDAVIT OF DAVID M. TUMEY
UNDER 37 CFR §§ 1.131 and 1.132

I, David Malcolm Tumey, being duly sworn, state as follows:

1. I am over 21 years of age and am competent to make this affidavit.

2. I graduated with a Bachelor of Science degree in Electrical Engineering from the University of Massachusetts in 1985.

3. I have close to twenty years experience working in the field of electrical engineering and electro-mechanical research and design, including extensive experience in the design and development of both hard-wired circuits and microcontrollers.

4. Between 1988 and 1992, I worked for the U.S. Air Force as an electrical engineer researching advanced cockpit design and aircraft avionic systems. I contributed to the development of neural-network based control systems for flight simulators that, by detecting a pilot's brain waves and learning associations between the brain waves and movements, enabled pilots to fly the simulator with their thoughts.

5. Since 1992, I have worked as an employee or consultant for various medical device firms helping them develop hardware and software control systems for sophisticated electromechanical medical devices.

6. I am listed as the sole or joint inventor of at least 32 issued U.S. patents and dozens of other foreign patents and foreign and domestic patent applications.

7. I and Dr. Tianning (Tony) Xu are the inventors of the subject matter set forth in U.S. Patent Application No. 09/488,390 entitled "Animated Toy Utilizing Artificial Intelligence and Facial Image Recognition."

8. The invention disclosed in 09/488,390 was a difficult and complex invention. We expended many years of effort developing the algorithms, methods, electrical circuits, and mechanical systems necessary to reduce the invention to actual practice.

Conception of the Invention

9. In 1989, I began sketching out the structure of an algorithm for recognizing facial images. *See* Exhibit "1." Early on, I realized that some of the biggest challenges I needed to overcome involved locating a human or human-like face in an image and scaling, rotating, and centering the face in an image window. *Id.*

10. Over the next several years, Dr. Xu and I began developing algorithms to preprocess the images. I developed a crude software program using WATCOM C++ to implement some of the algorithms disclosed in my Jan. 5, 1990 notebook entry. *See* Exhibit "2."

11. In a 1991 notebook entry I record techniques that we developed for finding a face in an image. *See* Exhibit "4." The algorithms involved complex matrix manipulations. Unfortunately, existing computer hardware technology at the time was simply inadequate to perform these manipulations in a time-, energy-, and space-efficient manner. We also began developing a library of functions to use for future prototypes.

12. As development continued, we identified other challenges to successfully reducing the face recognition concept to practice. *See* Exhibit "3." Aside from locating a face in an image, the system needed to be capable of determining whether a face even existed in the image. *Id.* Furthermore, the system needed to be able to filter the image to remove background information. *Id.*

13. Two notebook entries, one of which is dated Jan. 12, 1992, detail a method we developed for enrolling and creating templates for use in later recognition efforts. *See* Exhibits "5" & "6."

14. A Nov. 12, 1996 notebook entry records my conception of using the face recognition methods we previously developed to control the animation of a toy or video game or even to direct internet browsing. *See* Exhibit "7." In that notebook entry, I suggest using those methods not only for biometric facial identification, but also for facial expression recognition. I also noted that the combined biometric facial recognition and facial expression recognition could be accomplished by storing templates ("eigenfaces") of the same individual with different facial expressions.

15. These exhibits establish that we had possession of the whole invention claimed or something falling within the claim. Exhibit "7" shows entertainment devices, including toys, positionable in proximity to animate or inanimate objects. Exhibits "1" through "7" show possession of the image capturing and recognition elements (including a processor) of claims 1, 11, and 12. Exhibit "7" also shows possession of the concept of having the device provide entertaining interaction in response to recognizing a human face, and in particular, in response to recognizing a human face with a particular emotional expression.

Diligent Reduction to Practice

16. From September 1996 until the date we filed our provisional application, Dr. Xu and I worked diligently to reduce the invention to practice. Most of this time was spent on developing software algorithms to improve the performance of the face recognition technology. Because the effort was focused mainly on software, we have only a few engineering notebook entries to document the progress. We do, however, have several software programs, files, and data to corroborate the statements made in the following paragraphs. Attached as Exhibit "8" to this declaration is a four-page table listing the functions of several different versions of software programs and executables, along with the dates the files were last modified. The actual software files themselves are also included in a CD that accompanies Exhibit "8."

17. In addition, during this timeframe we simultaneously experimented with fingerprint technology that could be used in conjunction with the facial biometric as an adjunct or enhancement.

18. Face recognition algorithms that we had developed earlier for security applications were not fast enough to be used in a toy. They typically had about a 3-second delay between the image capture and the facial recognition – this delay was too long for a toy. Our goal was to make a toy that would recognize and respond to human users in real time.

19. Between September 1996 and February 1997, we began developing software for finding, aligning, normalizing and recognizing a unique face, plus detecting a not-necessarily-unique facial expression. During this time, using both Visual Basic and Visual C++, we began coding algorithms which could locate a face in a digital video

image. First, the video signal from a camera was continuously digitized and a number of single frame samples were stored in memory (samples that were discarded and replaced as the stream continued and computations were completed). The reason we used more than one frame was to enable the employment of motion detection/tracking algorithms. It was our thinking at the time, that the face would have relative motion to the background images. Thus, we utilized a subtraction algorithm which looked at the differences between two or more static images and located a region of motion where further analysis could be concentrated to detect a face. We also employed neural network algorithms for face detection, two dimensional FFTs and Hidden Markov Models. Ultimately, by using a blend of algorithms, we successfully solved the problem of finding a face in an image, framing it (drawing a circle or box around it) and tracking it as it moves. This legacy "box" is still used by the prototype today and can be seen in the demonstration video. *See* Exhibit "14". A red box indicates non-recognition of the face, while a green box indicates recognition. During this timeframe, we also began work on aligning and normalizing the located face (making all the detected facial images the same size – pixel area – and adjusting for lighting in the image – dark versus light).

20. Some of the software modules we generated during this time period include FR32.exe, FR32a.exe, FR32b.exe, FR32c.exe and CvidCap.dll – iterations of algorithms designed for capturing and processing facial images as described in Exhibit "8" attached hereto.

21. During this time frame, we also worked on the integration of a fingerprint sensor with an automobile key, as illustrated in the March 8, 1997 notebook entry attached as Exhibit "9". The key would have a capacitive sensor integrated in the grip,

electronic contacts and an electrical receptor which would be part of the automobile's ignition. A schematic diagram is provided which shows how the system would be used with a vehicle. During this time, work on the Face Recognition toy and the Integrated Biometric Key took place in parallel.

22. By the spring of 1997, we had developed a reliable solution for locating a face in near real-time. So our focus shifted to identifying individual features within the face to enable us to align the face better with the matching templates (eigenfaces). We first utilized neural networks, trained to recognize eyes, noses and lips (mouth) and also subsequently tested "eigenfeatures" for performing this localization. The purpose of this part of the algorithm was to locate each feature and find its position relative to the face; this enabled us to normalize for scale, translation (position) and rotation. It also helped the software determine if the image was actually a face (e.g., a face should have two eyes relatively above one nose relatively above one mouth).

23. Some of the software modules we generated during this time period include FR32d.exe, and VideoCap.exe – improvements which provided near real-time performance as described in Exhibit "8" attached hereto.

24. At this step, *feature* recognition was being accomplished and the computer could ascertain a person's *emotive facial expressions* with no further processing. But at this point, the algorithm *still has not uniquely identified the person.* In other words, additional processing was necessary, even after facial expression recognition, in order to provide biometric identification.

25. It must be understood that unique facial recognition is entirely distinguishable, both in the object sought to be achieved and in the methods used to

achieve it, from detecting a facial expression. Indeed, in our experience, unique facial image recognition is significantly more complex than detecting a not-necessarily-unique facial expression. Identification of the features only tells us that we have a face, maybe a smiling face or a frowning face. But we cannot know who's face we have until we perform the next steps in the algorithms of recognizing the combined set of features/expressions in the context of the whole (or Gestalt) face. This is an important distinction over Kamiya et al., which essentially stopped at the step of ascertaining a facial expression (and did not suggest or teach subsequent facial image recognition).

26. By the summer 1997, we had developed adequate algorithms to locate, track, align and normalize the faces. So our focus shifted to performing actual biometric identification.

27. One of the software modules we generated during this time period was FR32e.exe – an updated and improved algorithm which could identify facial features as described herein above and in Exhibit "8" attached hereto.

28. Biometric identification, which again is completely different from feature/expression recognition, involves treating the entire face as a unique collection of unique features that are specific to each given individual, i.e., the distance between the eyes, nose and mouth coupled with the morphology of each feature, make up a unique identification set much like the ridges in a fingerprint.

29. Our work centered on perfecting the "eigenface" approach (known formally as Principal Component Analysis). The eigenfaces represent orthogonal dimensions which collectively describe something we called "face space".

30. The first step in using the eigenface approach is to create this "basis" set of eigenfaces. We accomplished this task by first obtaining digital images from several hundred faces then culling through the photos to identify a "spanning" set of training images. By spanning we mean a set of images that contain as many of the feature variations we expect to encounter with the faces that will subsequently be recognized by the toy. Next, an average face is derived. Then, a standard method called the "Karhunen-Loeve Transform" is utilized to produce the actual eigenfaces.

31. By employing the eigenface technique, the complete face (unique collection of unique features), can be reduced to a small set of numerical "coefficients" that represent the location of that unique face in face-space. Thus, if six eigenfaces are employed, there will be six coefficients representing that face. Recognition is then performed by computing the Euclidean distance between subsequent measurements in face-space, e.g., the unique coefficients for a first face can be compared to the unique coefficients of a second face by measuring the distance between the two face's respective locations in face-space.

32. With the algorithm now working, we next began the tedious effort of tweaking parameters (how many faces in the training set, which faces in the training set, how many coefficients, etc.) to optimize performance.

33. Attached as Exhibit "10" is a notebook entry dated September 11, 1997. This second reference to the Integrated Biometric Key describes the use of a DTMF tone decoder to control the vehicle's ignition upon receiving a recognition signal. This tone decoder technology is very similar to that which was utilized with controlling the

animation features of the toy, and in this case the two different inventions shared similar hardware resources.

34. During the fall of 1997, we focused exclusively on optimizing the algorithm and experimenting with variations to improve performance. We learned during this time that we could reliably extract the facial expression information in combination with the biometric information to make determinations such as "Bob is frowning", and "Sally is happy". It was further discovered that the algorithms could easily recognize toys and inanimate objects as long as they also possessed a unique collection of unique features that had facial characteristics. We were able to get the technology to identify specific dolls as well as human users. <u>Since inanimate objects are incapable of producing spontaneous facial expressions, this is another example of the differences between this invention and the prior art. None of the prior art references suggest that the facial image of a doll could ever be recognized because the prior art does not complete this biometric recognition step. By contrast, this invention can and does recognize inanimate (no facial expressions) objects.</u>

35. One of the software modules we generated during this time period was Tone.wav – an early file used in testing and developing the use of audio control signals of an external device as described in Exhibit "8" attached hereto.

36. During the few first months of 1998, Dr. Xu and I worked on several different biometric inventions at the same time, including a fingerprint-key biometric system for which we filed a separate application for patent in 1999. All of our biometric projects had significant technological aspects in common with the toy project we were working on.

37. Attached as Exhibit "11" are photographs in a patent notebook entry dated January 5, 1998, of actual hardware built for integrating the Biometric Key with an Oldsmobile Silhouette. The system utilized a laptop computer which generated an audio signal upon recognizing the user. This audio signal controlled a relay which started the engine of the car when energized. Similar hardware was used for the face recognition toy. This entry shows the parallels in hardware between the two inventions which were being developed simultaneously.

38. Attached as Exhibit "12" is a patent notebook entry dated March 21, 1998, showing excerpts of C++ software utilized in implementing the key system. We used similar modules for the toy project. There were several shared functions between the systems, especially as described above with respect to the audio signals controlling the hardware subsequent to the recognition process.

39. By the summer of 1998, our work associated with acquiring live facial images and processing and recognizing them was largely complete. Our focus shifted to developing a working toy demonstration which would use an off-the-shelf platform (Teddy Ruxpin was chosen) that could speak, articulate, recognize and interact with a human user. In addition to the ability to recognize faces, additional neural networks were developed that permitted the toy to recognize numbers, letters and shapes. That way, the toy could be utilized for various learning applications. For example, the toy would know that it is playing with "Billy" who is six years old and learning shapes. If Billy drew a box, it could say "nice job Billy". If Sally came in the room, the toy would immediately recognize her and now realize that it was playing with two children and modify its interactive behavior accordingly.

40. Most of our time from June to August of 1998 was spent in understanding the control method for the toy. We used two tracks of audio (left and right). One track produced the audio (speech) you hear from the toy, and the other controlled the movement of the mouth and eyes.

41. I had to develop a multi-media workstation that enabled me to record my voice along with a control signal (produced by moving a joystick) in time with my speaking so that the bear could be properly animated. Next, we stored the combined audio and control signals in a .wav file which was later played by the software when the toy was interacting with a human user. The .wav file selection was done using decision tree logic. This was a difficult and time consuming task. From August through October of 1998 we developed most of the demo media scripts using this workstation. By the end of October, the first "bench-top" prototype system was ready for integration.

42. With the earliest prototype beginning to function as intended, we made various refinements between October 1998 and December 1998, including incorporating the camera into one of the bears. (Most demos are done with an externally mounted camera as seen in the video). In addition, we obtained newer model Ruxpins that were more reliable in their operation.

43. By early December, 1998, we finally completed a working prototype of a stuffed bear toy with facial biometric recognition capabilities and which provided entertaining interaction in response to recognition. By this time, we had developed and successfully tested the following functionality, including: (1) the ability to locate and track a face in a video image; (2) the ability to scale, normalize, rotate and translate a facial image; (3) the ability to recognize and locate facial features (eyes, nose, mouth)

within the face further refining the alignment of the face sufficient for Principal Component Analysis, and (4) biometrically recognizing individual faces which consisted of a unique collection of unique features for a number of different individuals playing with the toy. In addition, we integrated the toy with a PC through the use of the sound card that enabled articulated interaction with the human user through the use of .wav files which consisted of an audio track and control track.

44. Some of the software modules we generated during this time period include Bubba.wav, Dolly.wav, Happy.wav, Sad.wav, and Woody.wav – animation and script files used to create the toy interaction as described in Exhibit "8" attached hereto.

45. Attached as Exhibit "13" is a December 20, 1998 entry that outlines the ideas which we incorporated in the Face Recognition Toy Provisional Patent that we filed in January, 1999. It discloses the bear, the animation and techniques for accomplishing it, and the various games and applications that can be implemented utilizing the facial biometric technology. All of the elements of the present invention are discussed in this entry, and at this time or very shortly thereafter, a final working prototype was produced that functioned very well and demonstrated a complete reduction to practice. A video of one of these demonstrations was made and is available on the CD attached as Exhibit "14". All demonstrations of the toy to outside parties were made after the filing of the provisional application.

46. We began working on drafting the provisional application in early January 1999. We filed the provisional application on Jan. 19, 1999.

47. I acknowledge that willful false statements and the like are punishable by fine or imprisonment, or both, and may jeopardize the validity of the application or any patent issuing thereon.

48. I hereby declare that all statements made of my knowledge herein are true and that all statements made on information and belief are believed to be true.

David M. Tumey

STATE OF FLORIDA §
COUNTY OF Okaloosa §

BEFORE ME, the undersigned authority, on this day personally appeared DAVID M. TUMEY, known to me to be the person of that name, who signed the foregoing instrument, and acknowledged the same to be his free act and deed.

GIVEN under my hand and seal of office this 14 day of March, 2005.

Notary Public

TIFFANY JOHNSON
MY COMMISSION # CC 327908
EXPIRES: June 10, 2008

Tiffany Johnson
Printed Name of Notary

Commission Expires June 10, 2008

AFFIDAVIT OF TIANNING XU

I, Dr. Tianning Xu, being duly sworn, state as follows:

1. I am over 21 years of age and am competent to make this affidavit.

2. I have read Dr. Tumey's affidavit and hereby affirm that the statements made therein are true and consistent with my own knowledge, beliefs, and recollection.

Dr. Tianning Xu

STATE OF TEXAS §
COUNTY OF BEXAR §

BEFORE ME, the undersigned authority, on this day personally appeared DR. TIANNING XU, known to me to be the person of that name, who signed the foregoing instrument, and acknowledged the same to be his free act and deed.

GIVEN under my hand and seal of office this 10 day of March, 2005.

COURTNEY D. OSGOOD
NOTARY PUBLIC
STATE OF TEXAS
My Comm. Exp. 05-06-06

Courtney Osgood
Notary Public

Courtney Osgood
Printed Name of Notary

Commission Expires 05-06-06

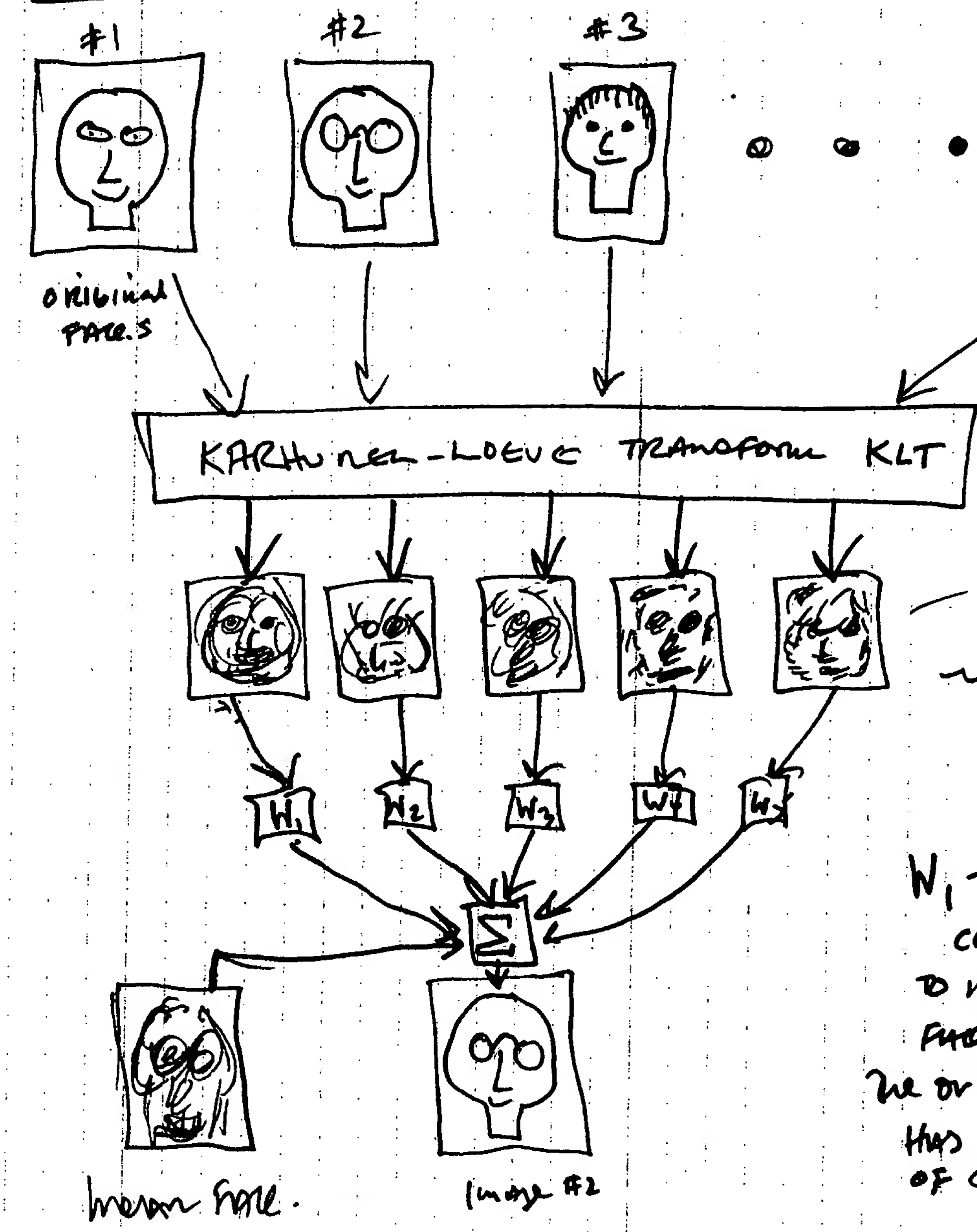
FACE RECOGNITION USING EIGENVECTORS

21 NOV '87 Dent

* meeting w/ CRAIG Arendt

overall concept:

FACE PIXEL SPACE.



First 5 eigen vectors

~ store in the CPU or main database

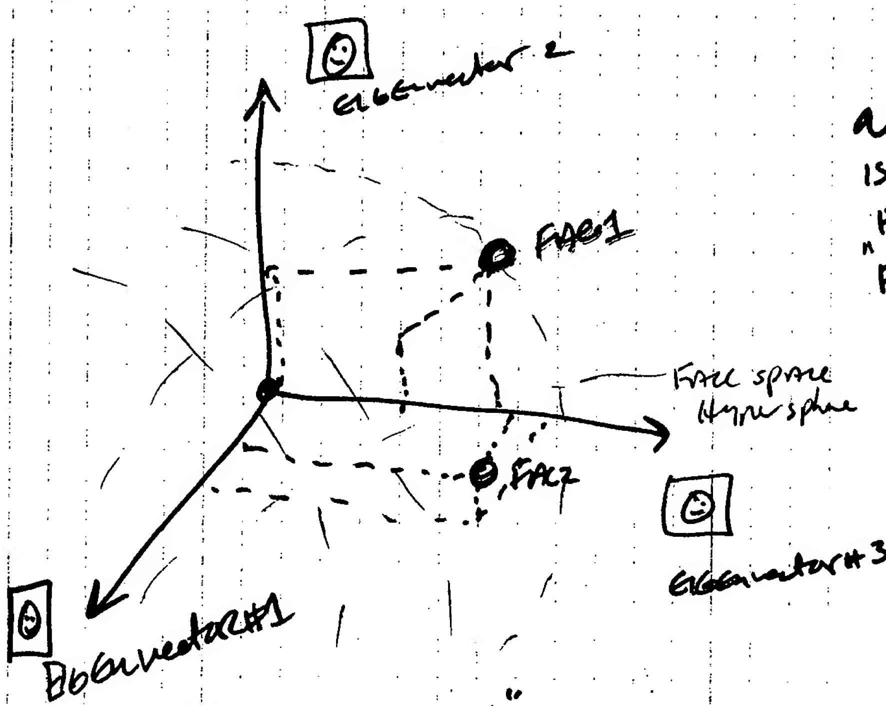
$W_1 - W_5$ = coefficients to reconstruct original face. Each of the original faces has a unique set of coefficients

[redacted]

FACE RECOGNITION CONT'D

Dmt 11/87

✗ Visual Representation of How it works:



Assume each Eigenvector Face is orthogonal therefore Fig 1 is a 3 dimensional "FACE" space.

Fig 1 - "FACIAL" SPACE

Each FACE (1, 2, 3) can be described by its location in FACE SPACE by the Eigenvalues (x, y, z).

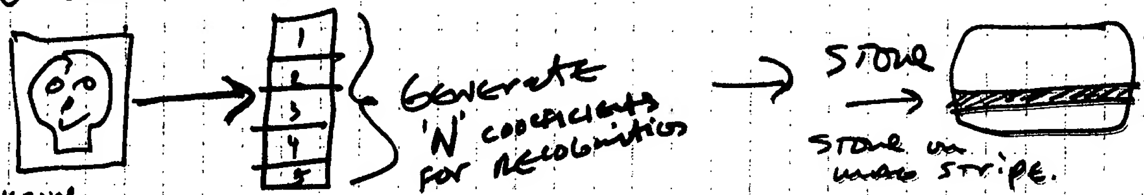
Therefore Face 1 can be described by (x_1, y_1, z_1) and face 2 as (x_2, y_2, z_2). If therefore each face is different and has different face space coordinates this technique will be extremely valuable in applications requiring secure identification of the user such as secure area access, ATM machines, teleconferencing, financial applications.

[redacted]

FACE REC con't

1/89 Draft
CRAIG Arndt.

① NEW user



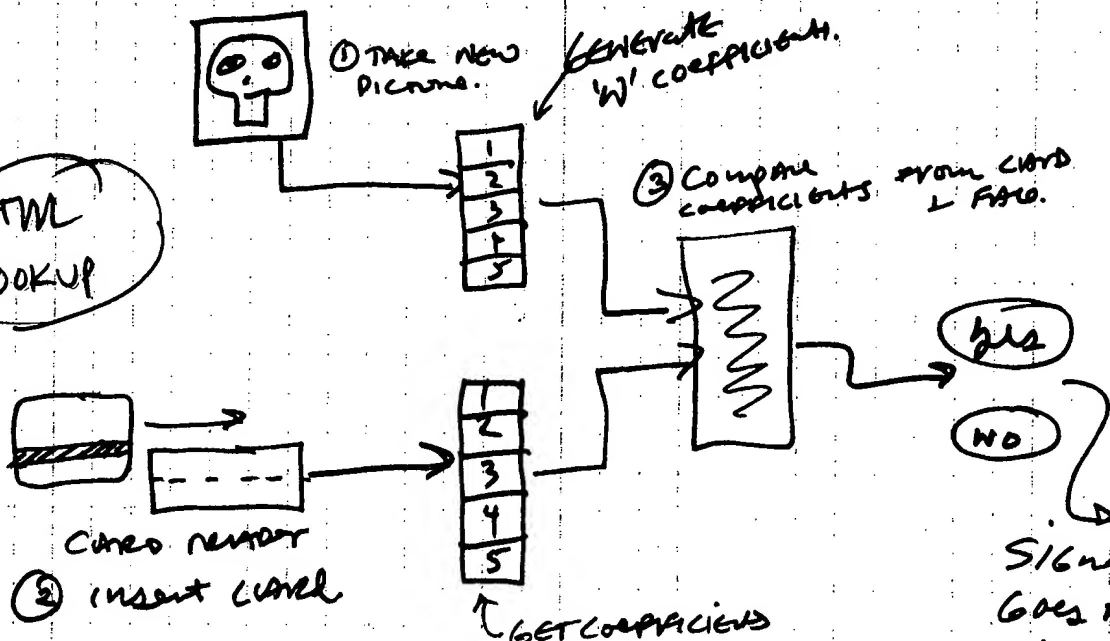
Picture taken

(Video digitized image)

② USER INTERFACE.

ATM HOOKUP

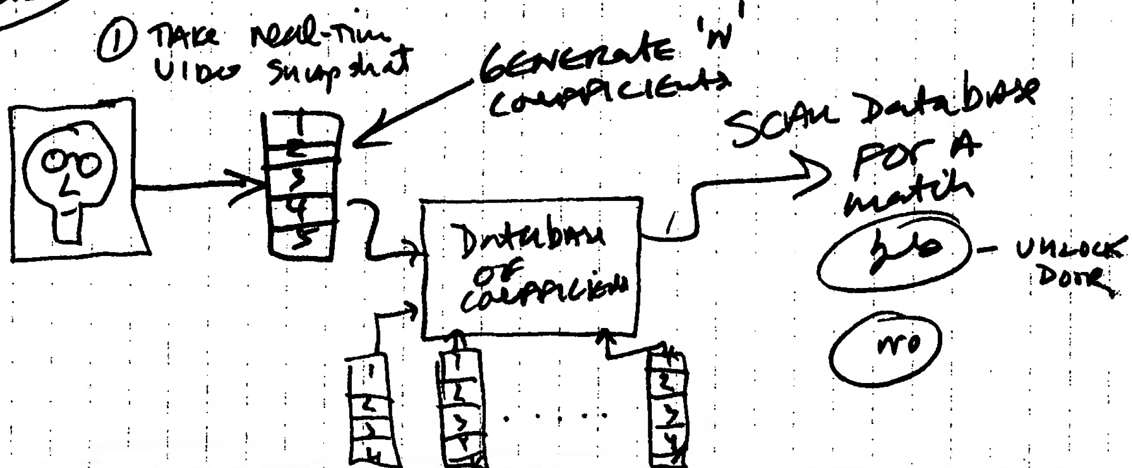
Fig 2



Signal goes to ATM machine for pass/fail

Secure Access

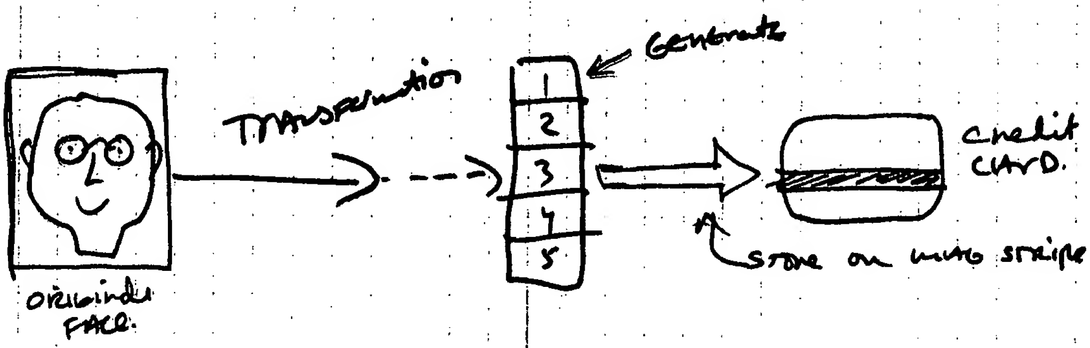
Fig 3



[redacted]

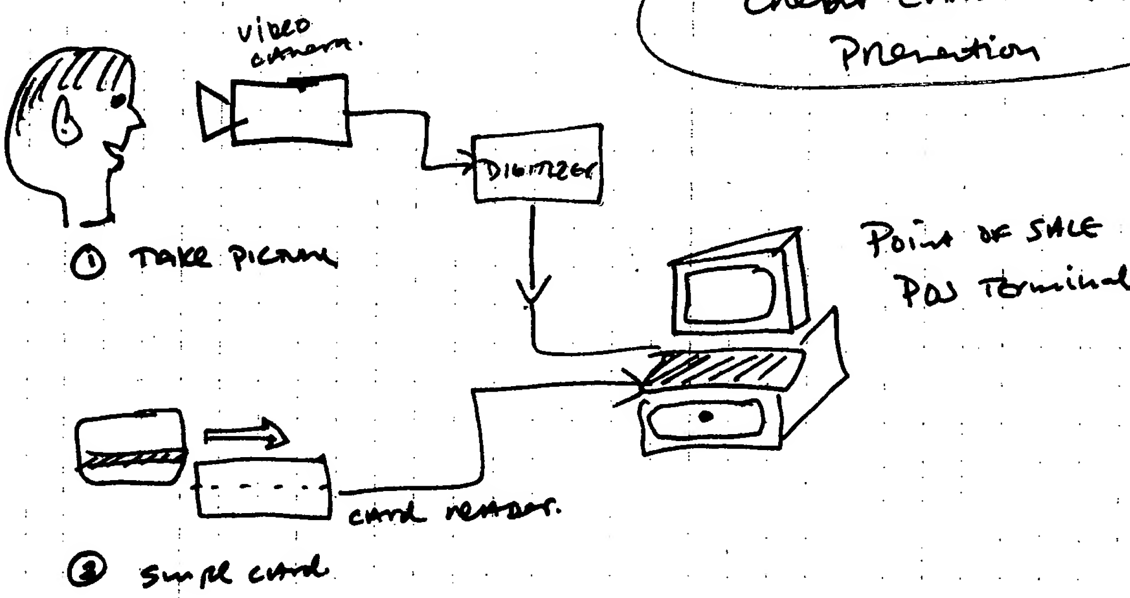
FACE RECOGNITION — CREDIT CARD Dut

FIG 1



USE

Credit card fraud prevention



STEP ① Ring up purchase.

② TAKE PICTURE w/ VIDEO CAMERA

③ SWIPE CARD → GET ACCOUNT INFO + COEFFICIENTS

④ a) visually reconstruct FACE from coefficients + show on screen

b) compare FACE + FACIAL COEFFICIENTS to verify user.

⑤ GET Authorization code + complete TRANSACTION.

[redacted]

Eigenvector FACE Recognition COTD 1/87 Dut

Visual IMAGE preprocessing

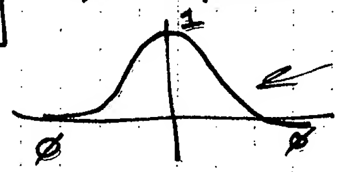
Problems: scaling → How Big/Small the FACE IMAGE Actually is

ROTATION → FACE TILTED from vertical

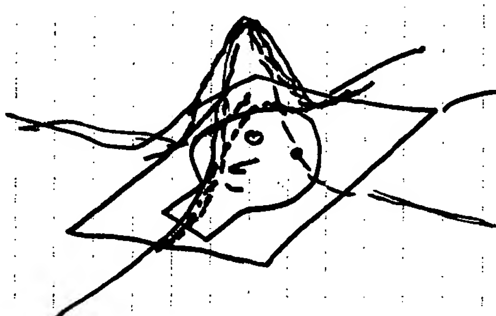
TRANSLATION → FACE NOT CENTERED in WINDOW

BACKGROUND DISTRACTION → OBJECTS in BACKGROUND

$\boxed{\text{BACKGROUND DISTRACTION}}$ → GAUSSIAN WINDOWING (2 dimensional)



← GAUSSIAN Function (1 Dimensional)



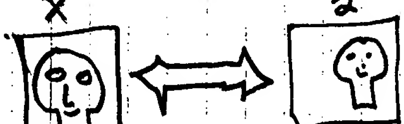
pixel space.

Multiply each pixel in the space By the GAUSSIAN Function so that centralized pixels are not attenuated as much as background pixels

FACE CENTERING + SCALING - Template matching

$\vec{X}$ $\vec{Z}$

First pixel $\vec{X}_1$



FACE Template

unscaled translated FACE

→ TAKE MSE Between two Faces

$$\sqrt{\sum_{i=1}^{N^2} (\vec{X_i} - \vec{Y_i})^2}$$

Then ADJUST FACE position for minimum error.

NxN pixel image Repeat For scaling

[redacted]

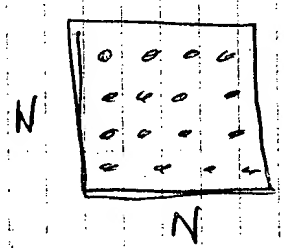
AUTOMATIC FACE RECOGNITION

5 JAN 90 [signature] inventor

EQUATIONS TO GENERATE THE EIGENVECTORS + EIGENVALUES ~~For~~ A member POPULATION OF FACES:

① GET MEAN FACE

$\vec{X}$ = input image



$n \times n$ pixel image

$\vec{X} = \{X_1, X_2, \ldots, X_{n^2}\}$

$$M_x = \text{mean face} = \frac{1}{M}\sum_{i=\emptyset}^{M} \vec{X}_i$$

M = TOTAL # OF FACES IN TRAINING SET.

$\vec{X}_i$ = individual pixel images OF FACES

M_x = mean FACE.

---

② Generate COVARIANCE MATRIX $\qquad$ M = # OF IMAGES

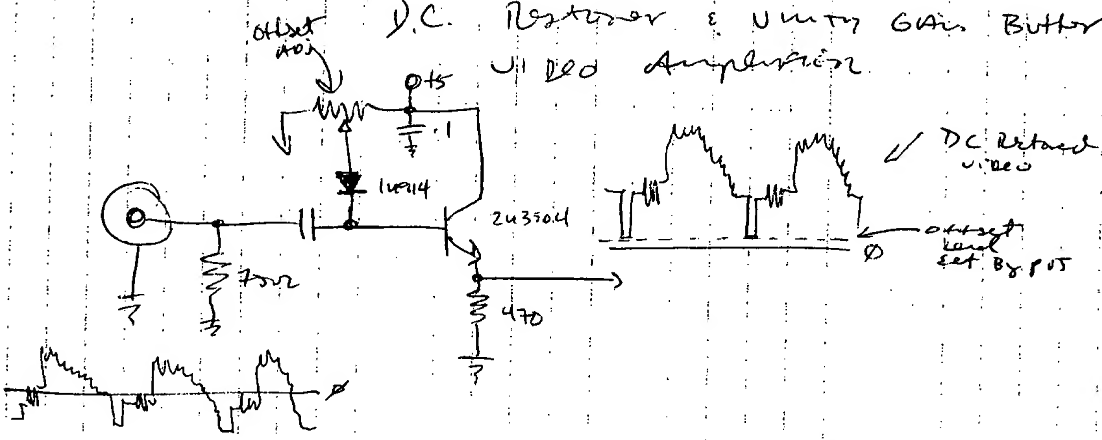
$$C = \frac{1}{M}\sum_{i=1}^{M} (\vec{X}_i - M_x)(\vec{x}_i - M_x)^T$$

This ~~compares~~ finds the covariance between The First pixel w/ each pixel of all images M

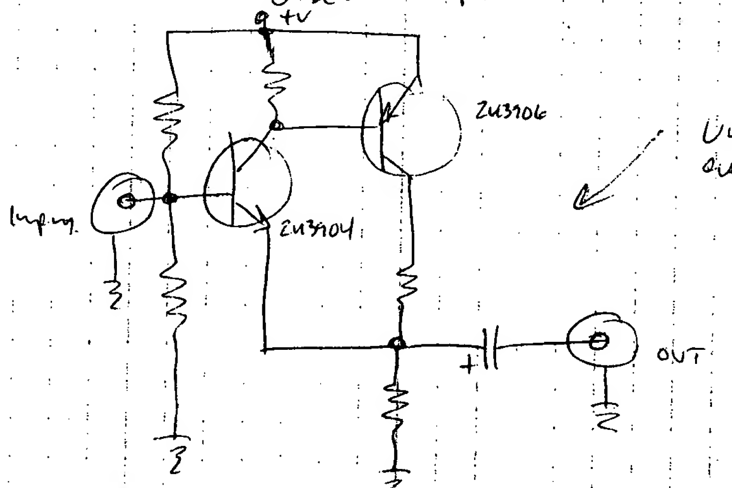
↳ results in A $n^2 \times n^2$ matrix

$$C = \frac{1}{M}\begin{bmatrix} \sum_{i=1}^{M} x_1^i x_1^i & \cdots & \sum_{i=1}^{M} x_1^i x_{n^2}^i \\ \vdots & & \\ \sum_{i=1}^{M} x_{n^2}^i x_1^i & \cdots & \sum_{i=1}^{M} x_{n^2}^i x_{n^2}^i \end{bmatrix} - \vec{M}_x M_x^T$$

D.C. Restorer & Unity Gain Buffer Video Amplifier



Video Amp.



Unity Gain Buffer capable of driving 50–75 Ω loads

③ Find Eigen vectors & Eigen values of covariance matrix C & sort in order of highest to lowest energy.

Results in N^2 eigenvectors & eigenvalues.

④ Truncate the # of eigenvectors & eigenvalues by looking at the mean sq error between an image & its reconstruction:

if e = eigenvalue then

$$MSE = \sum_{i=1}^{N^2} e_i^2 - \sum_{j=1}^{K} e_j^2$$

where K = # of highest energy eigenvectors chosen.

⑤ ~~Generate matrix~~ Generate matrix of the K largest eigenvectors.

$$\vec{B}_K \text{ Basis set (truncated)} = \begin{bmatrix} e_{11}, e_{12}\, e_{13}, \cdots e_{1N^2} \\ \vdots \\ e_{K1} \cdots e_{K2}\, e_{K3} \cdots e_{KN^2} \end{bmatrix}$$

Each eigenvector is a $N \times N$ image similar to the input image

⑥ Transform an original image $\vec{X}$:

$$Y_K = \vec{B}_K (\vec{X} - \vec{m}_X) \quad \text{for } K = 1 \text{ to } \# \text{ of eigenvectors}$$

y_K = the Kth coefficient for eigenvector(K)

$\vec{B}_K$ = the Kth eigenvector from the truncated matrix

$\vec{X}$ = original image

FACE REC —

(7) TO reconstruct the original image $\vec{X}$ from the y_K Eigenvalues simply:

$$\tilde{X} = \left[\sum_{K=1}^{M} \vec{B}_K^{T} y_K\right] + \vec{m}_x$$

M = # OF TRAINING images

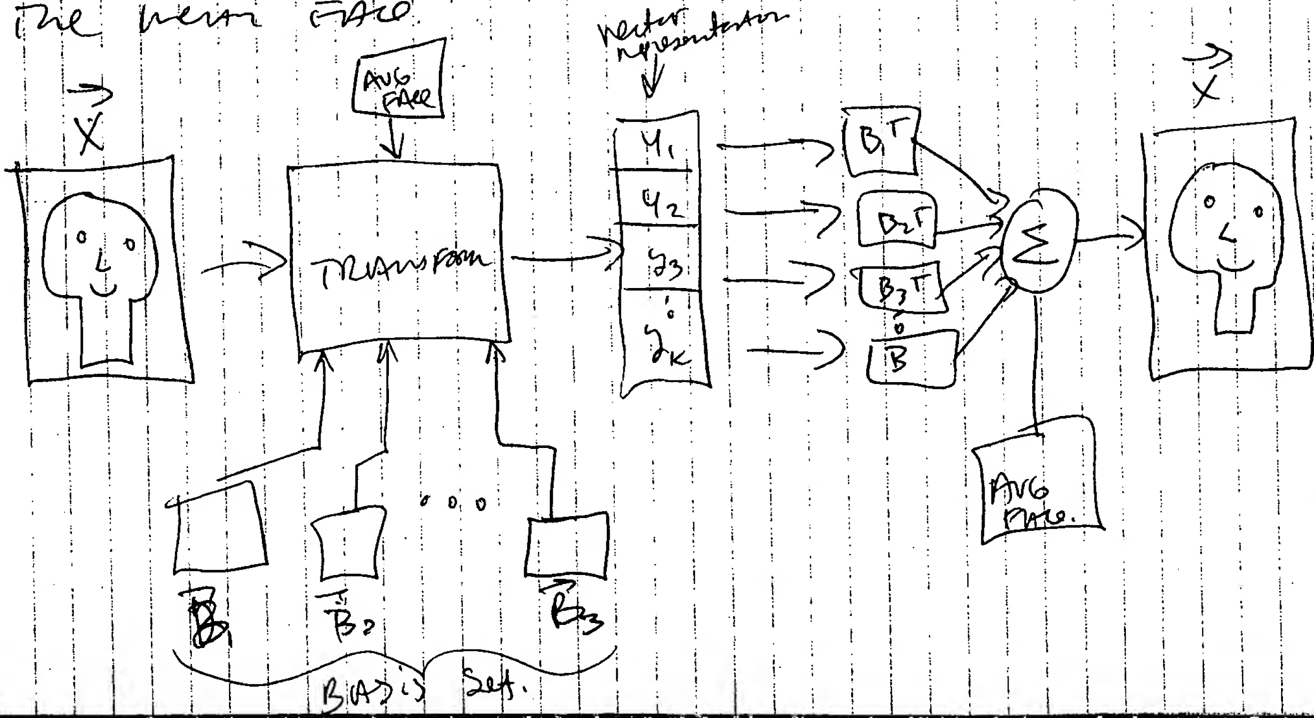
$\vec{m}_X$ = mean FACE.

$\vec{X}$ = original image.

$\vec{B}_K$ = BASIS set.

y_K = EIGENvector coefficients

TAKE DOT product of $B_K^T y_K$ then ADD BACK the mean FACE.



FACE RECOGNITION PROJECT / JUNE 27, 1991

Inventors: D. Tumey
C. Arndt
A. Aboujaoude (DEBAW)
T. Shue

Meeting Ris
Craig Insley

↳ patent issues
↳ should we file?

copyright
trade secret
multiple patents?

→ FACE RECOGNITION / VERIFICATION - ISSUES.

(1) LOCATE FACE - IS IT ACTUALLY A FACE - USE N.N. OR LOCATE EYES, NOSE, MOUTH w/ NEURAL NETWORKS + check relative positions. IF correct then you have A FACE.

Kurt
298-7116

(2) Find Eyes -
a) Neural network trained to find Eyes.

b) 2 dimension surface minimization to locate Eyes & mouth.

bread
milk
cooking
oil

(3) GAUSSIAN WINDOWING TO remove Background information

→ FACE SEGMENTATION
a) Neural network - Best Fit?
b) Digital subtraction of Background images.

(4) Scaling, ROTATION, DISTORTIONS such AS MOUSTACHE, EARRINGS, HAIRSTYLE, GLASSES.

D. Tumey
Witnessed: [illegible] 6-27-91

Center of mass II
(continued)

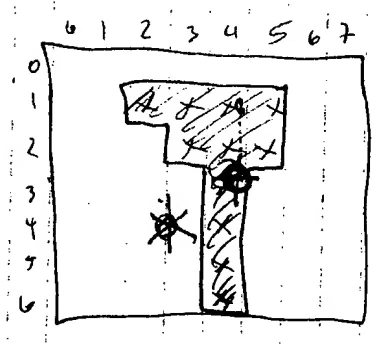
find C_mass (x_1, y_1)

x_1, y_1 = center of mass for this

```
for (i = 0; i < count; i++) {
    IF (HIT_BOX[i] == 0) Break;
    y1 += (int)(HIT_BOX[i]/128);
    x1 += (int)(HIT_BOX[i] - (y1 * 128));
}
```

i*128+j

```
x1 = x1 / count;
y1 = y1 / count;
```



1, 2
1, 3
1, 4
1, 5
2, 3
2, 4
2, 5
3, 4
4, 4
5, 4
6, 4

2.6 $\frac{28}{11}$ 42/11 39

11)280

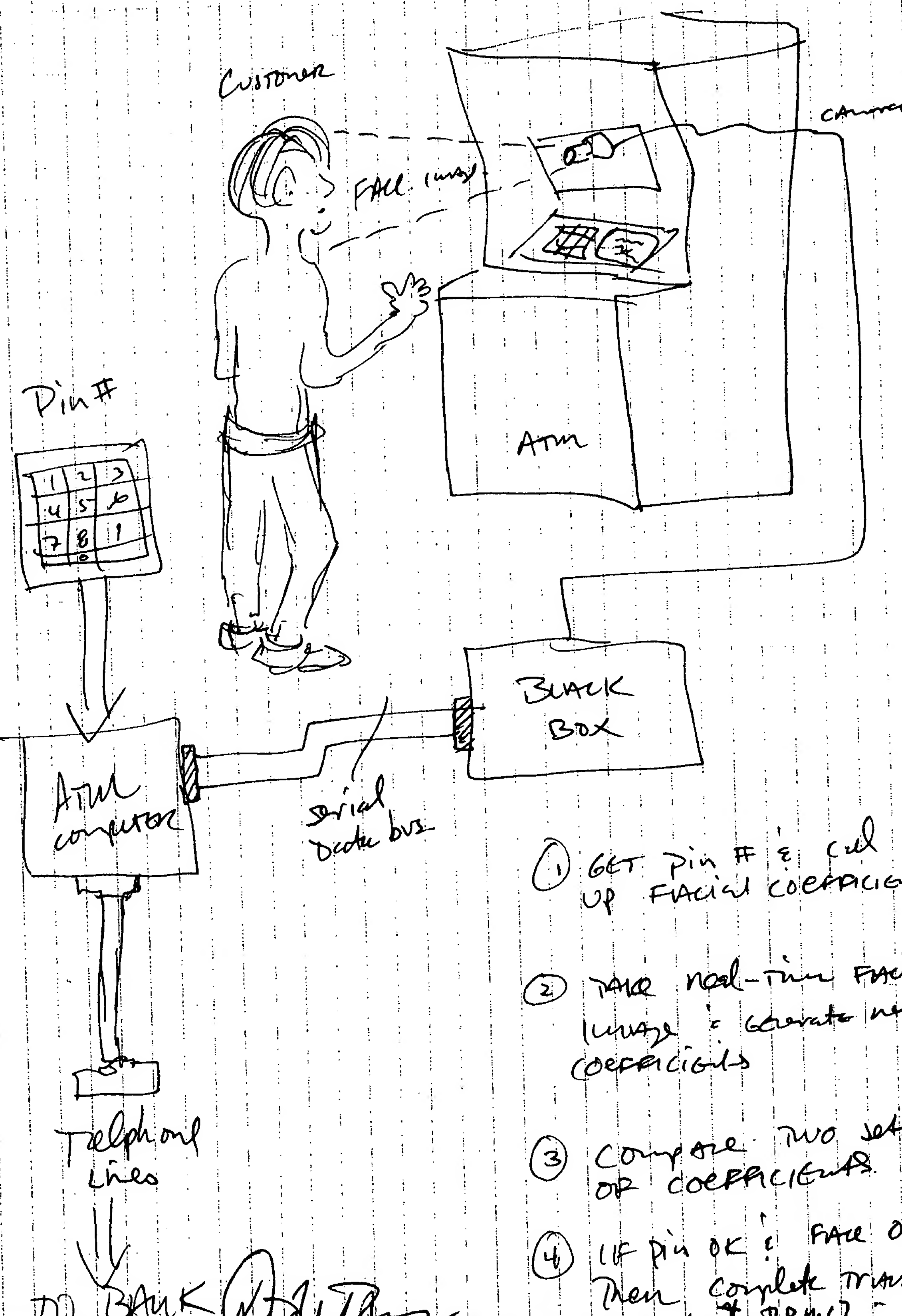
11
26
66
22x
286

11)2.6
220

3.9
11)420
330
90

FOOT PUMP APPLICATIONS 6/9/

ATM SYSTEM.



(1) GET Pin # & call up FACIAL COEFFICIENTS.

(2) TAKE real-time FACIAL image & generate new coefficients

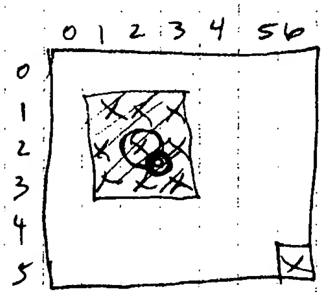
(3) COMPARE TWO sets OF COEFFICIENTS.

(4) IF Pin OK & FACE OK Then complete transaction

Wit. DBM 6/22/91

Center of mass I

face in frame (face detector)



X = HIT

☐ = miss

HIT LIST

1 , 1

1 , 2

1 , 3

2 , 1

2 , 2

2 , 3

3 , 1

3 , 2

3 , 3

$18/9 = 2$ $18/9 = 2$

2 , 2

6 , 5

$24/10$ $23/10 = 2.3$

2.4

Define array Hit_Box[ ]

AVG Pixel Location

for (i=0; i<(Row*Col); i++)

IF (~~Pixel~~ tr_image[i] > Thresh)

~~HIT-BOX[i] = i;~~

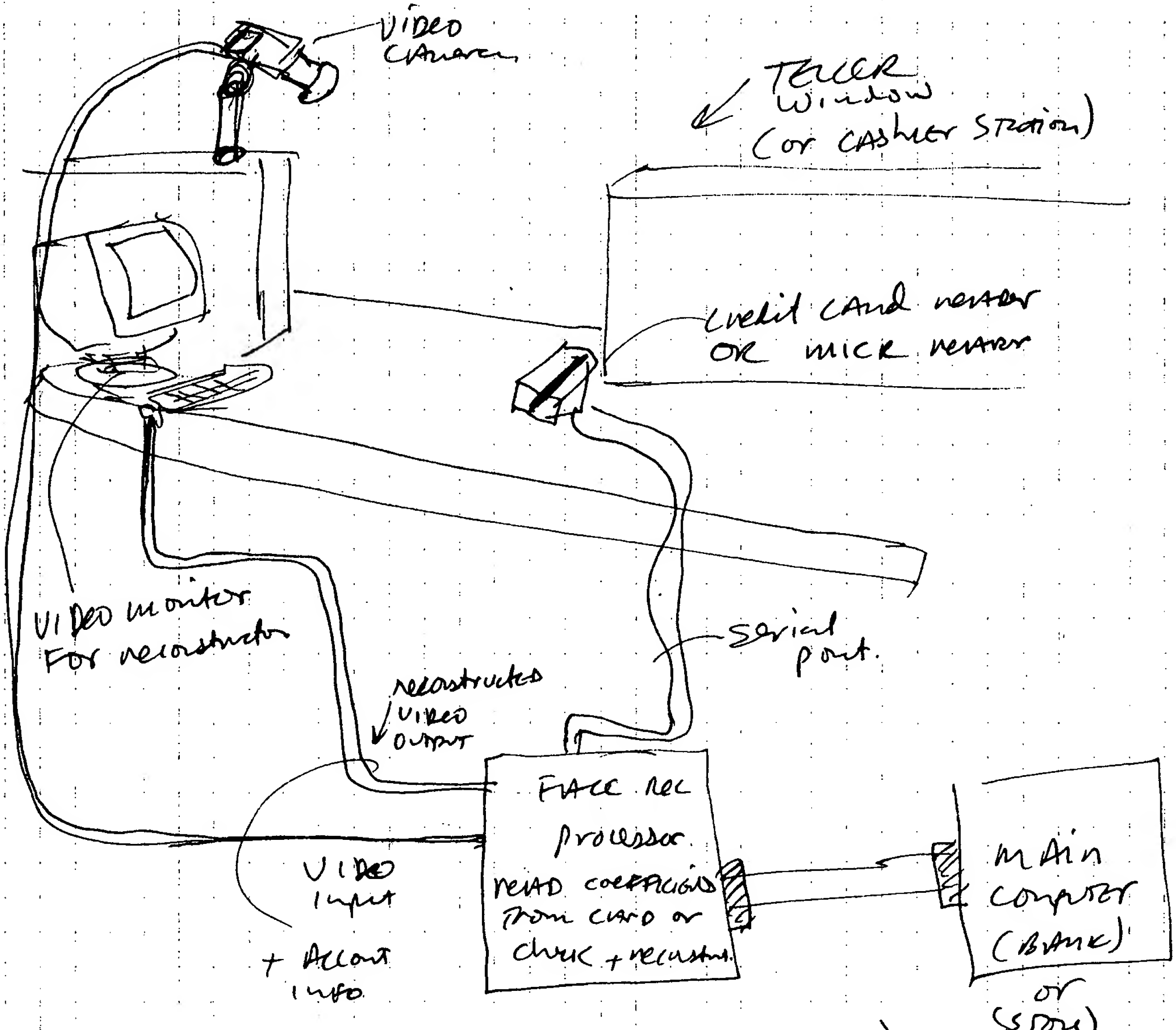
~~count~~ ~~count~~

→ HIT_BOX [count++] = i;

Loads HIT BOX array.

FACE REC CONT'D APPLICATIONS 6/91

CHECK CASHING / Credit card verification



① read card (magnetic) or check (MICR) & GET COEFFICIENTS.

② TAKE real time snapshot & Generate new set of coefficients

③ DISPLAY reconstructed FACE OF person on monitor

④ compare coefficients.
match if Automatic OK & visual OK

KDM 6-27-91

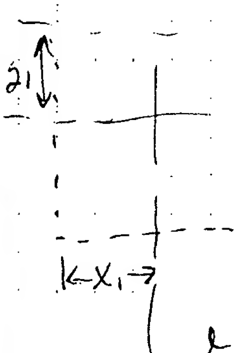
Finding Face in An image III

Center of mass calculations

TAKE BACKGROUND image → TAKE real-Time image → Subtract images

CODE:

```
count = ∅;
for (i=16; i < Row; i++)
    for (j = 5; j < col ; j++) {
     IF (ABS( TR_image[i*128+j] ) > Thresh )
            HIT_BOX[count++] =(i*128+j);

for (i=∅ ; i< count; i++) {
    y1 += (int)( HIT_BOX[i]/128);
    x1 += (int) ( HIT-BOX [i] - (y1*128));
}

x1 = x1/count;
y1 = y1/count;

/** x1 & y1 = Center of mass OF Face **/

for(i =∅; i< Row*col; i++)
      ~~HIT-BOX[i] = ∅;~~

~~for~~ IF( ( i>(~~50-X1/2~~ x1 - X-BOX/2) && i < (x1 + X-BOX/2)
  && { y1 >( y1 - Y-BOX/2 ) && y1 < (y1 + Y-BOX/2))
           HIT_BOX[i] = ~~TR~~_image[i];
  Else  HIT_BOX[i] = ∅;
```



rename HIT_BOX = FACE_BOX

for (i=∅; i< Row*col; i++)
~~HIT~~ P_Value) FACE_BOX.)

VID TO - ASC + 256-128 Video frame conversion Routines.

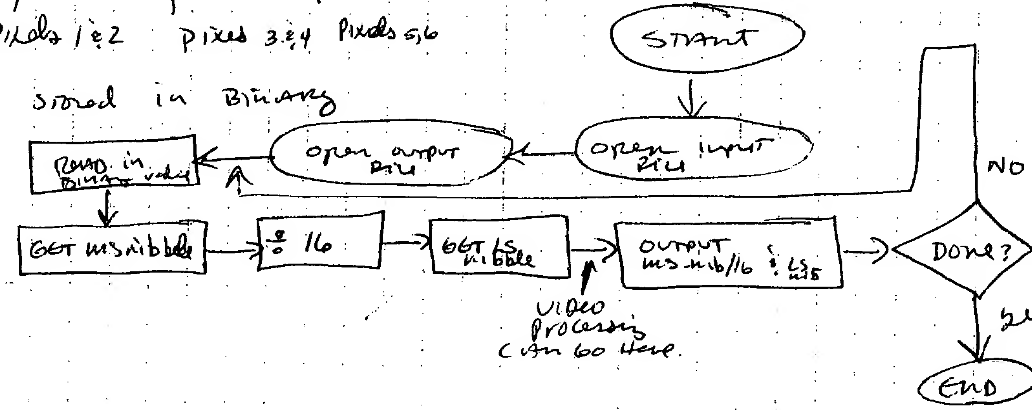
VID_ ASC - converts Image ~~stored~~ Files stored in the DAK. VID Format

which is: $\boxed{A|3}$ $\boxed{F|1}$ $\boxed{B|2}$ ∘ ∘ ∘ etc. Image size is 254 x 232

BYTE1 BYTE2 BYTE3

Pixels 1 & 2 pixel 3 & 4 Pixels 5,6

Files are stored in Binary



Converting 256 x 256 into 128 x 128 pixel images.

① # define N 65536 (255 x 255)

② allocate memory for 2 Arrays as ① Image [65536] ② trunc_image[16384] (128 x 128)

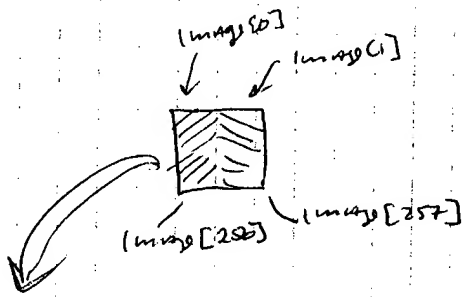
③ open "infile" And "outfile"

④ Clear out image[] array   for (k=ø; k<N; k++)
                                    image [k] = ø;

⑤ read in 256² image

```
for(k=ø; k<n; k++){
    fscan(infile, "%d", &f_buf)
    image [k] = f_buf;
}
```

⑥ Truncate original image By Pixel Averaging:




```
y=1;
x=ø;
for (i=ø; i<128; i++) {
    for (j=ø; j<128; j++){
        Temp = image[x] + image[x+1] + image[x+256] + image[x+257];
        IF (TEMP != ø) Temp = TEMP/4;
        tr_image[y] = Temp;
        y++;
        x+=2;
    }
    x+=256;
}
```

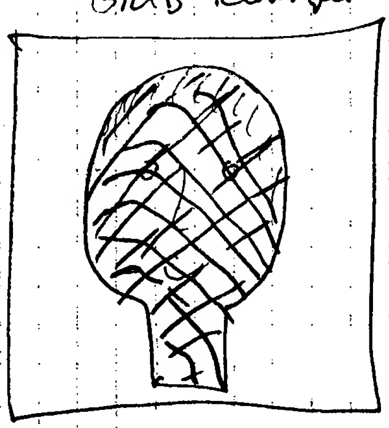
⑦ output truncated File.

19 Oct 92

Locating a Face in an Image

- Calculate distance (ultrasonics) + adjust zoom?
- Project 1cm² grid on face to find facial edges.
- Motion detection (Tom) ± subtract background.
- Dot technique where subject looks at dots; only a critical amount will allow them to be seen.

GRID technique



Subtraction

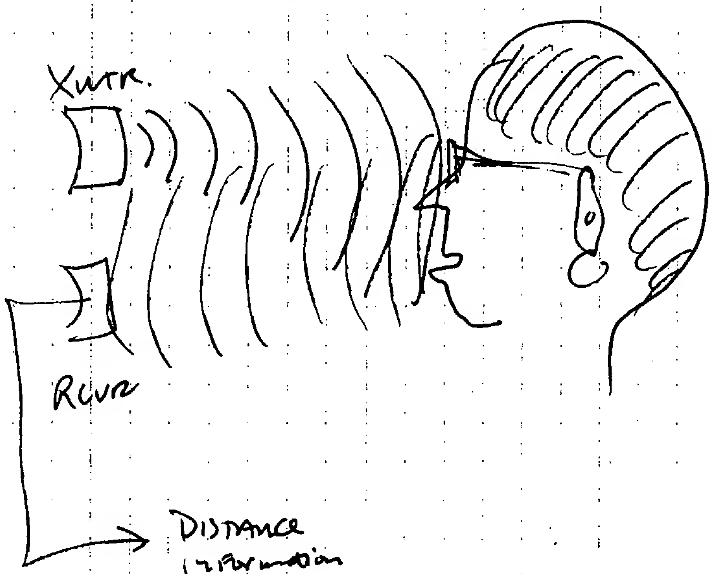
subtracted OUT

① Grid only illuminates face.

② Grid could be colored light ∴ camera detects a specific color.

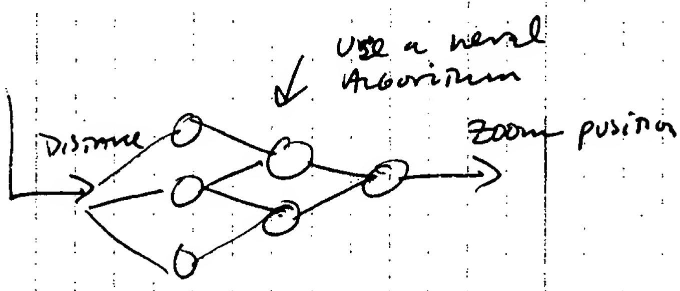
Remove background by subtracting one image from another + keeping the pixels from 2nd image (w/ face) based on |Δpixel| values.

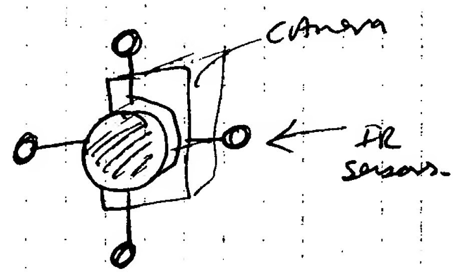
USING ULTRA SOUND TO ADJUST FACIAL SCALING VARIATIONS



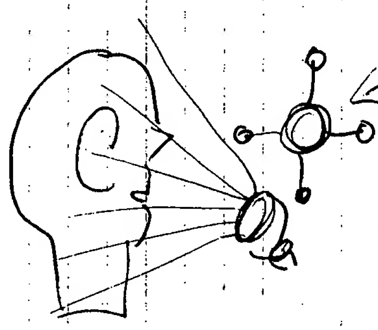
USE 4 SENSORS TO FIND FACE. + IR ILLUMINATION

ADJUST ZOOM





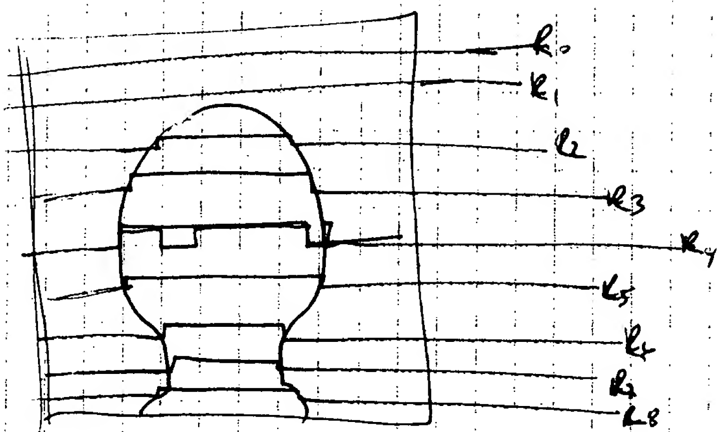
N.N. ZOOM CONTROL BASED ON ULTRASONIC INFO (DISTANCE TO FACE)



Inventor: 10/12/98

[redacted]

BACKGROUND SUBTRACTION

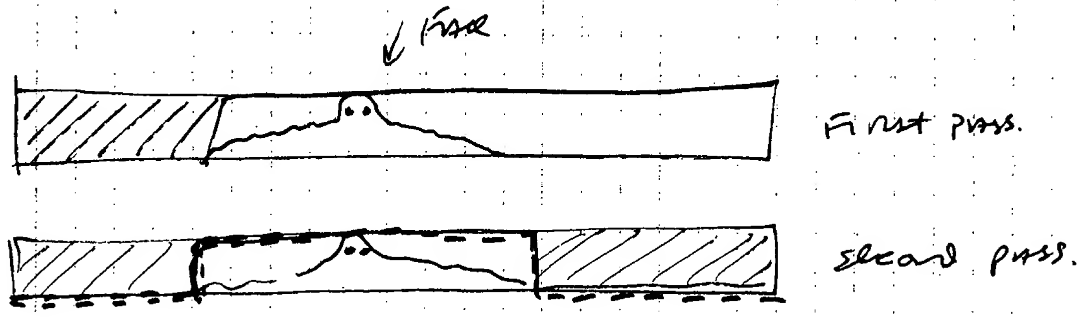


Each Row has only 2 on/off transitions
This preserves facial features that might otherwise be lost if the pixel levels are identical to background

C = (Pixa - Pixb) IF C < Thresh

```
for( i=0 i ≤ 128 , i++)
    C= [image_1 (i) - image_2(i)]^2
    IF( C  < Threshold &) frame_edge = false
                          image_2 (i) = 0 ;
    Lead Edge = TRUE; /* Disable above test */

for( i= 128; i=0; i--)
    C= [ image_1 (i) - image_2 (i))^2
    IF (C < Threshold & trail EDGE = FALSE)
        image_2 (i) = 0 ;
    Trail EDGE = TRUE;
```

for Any Row : Therefore,



Repeat for all Rows. The net result is the subjects face-only stored in image_2[i][j].

Step 2 — Scale Face

Step 3 — Using a correlation technique translate the face horizontally & vertically within the pixel image

Step 4 → Gaussian Window

Step 5 → Pass to Eigenfaces.

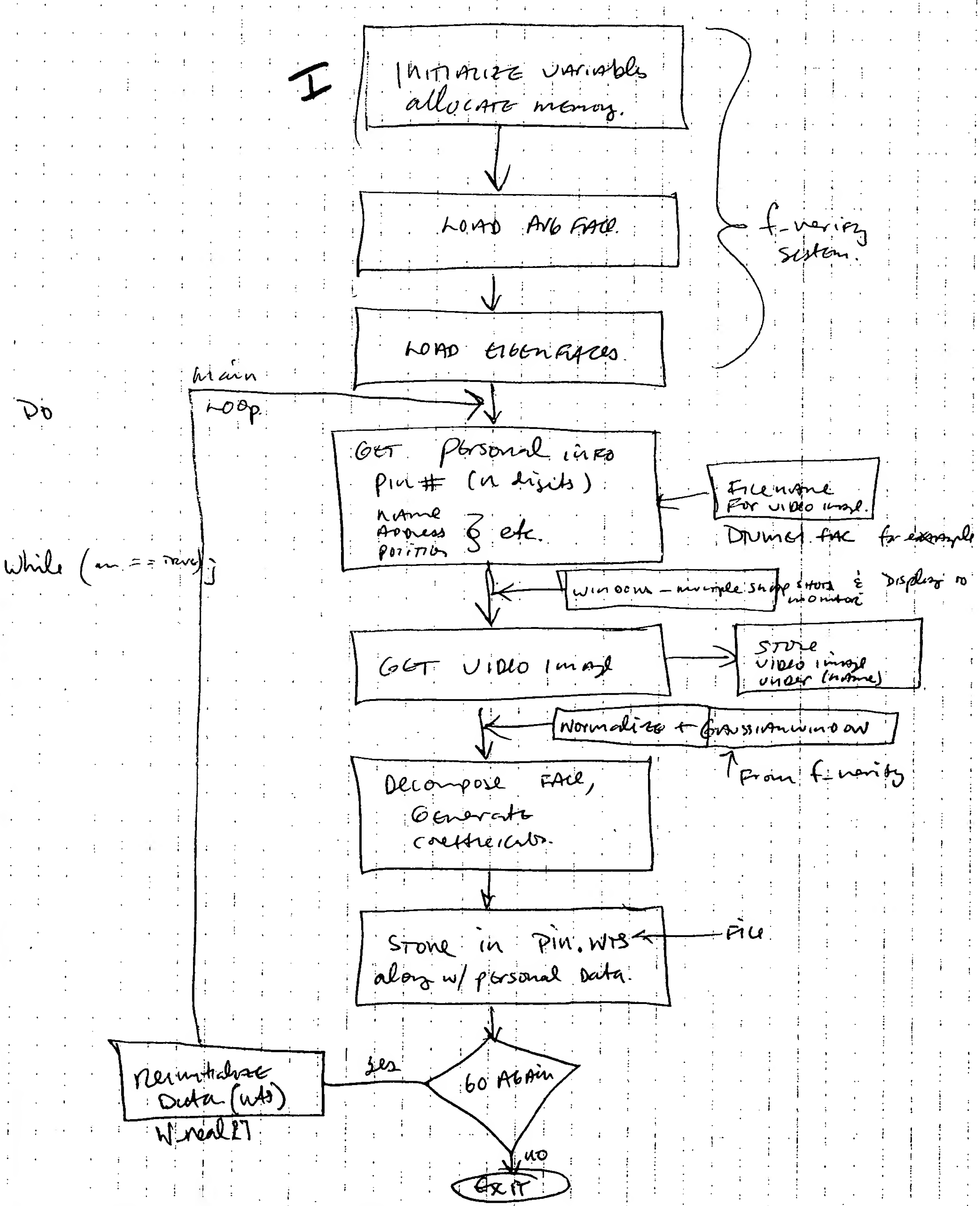
Step 6 → Periodically Test & Automatically update background file image.

Auto-Test
① Take 2nd background image. ~ or wait until no motion is present for period t > N sec.
② subtract from first image.
③ IF total pixel's of subtracted images < Thresh
OK - coast is clear.
④ Update BKG image.

[redacted]

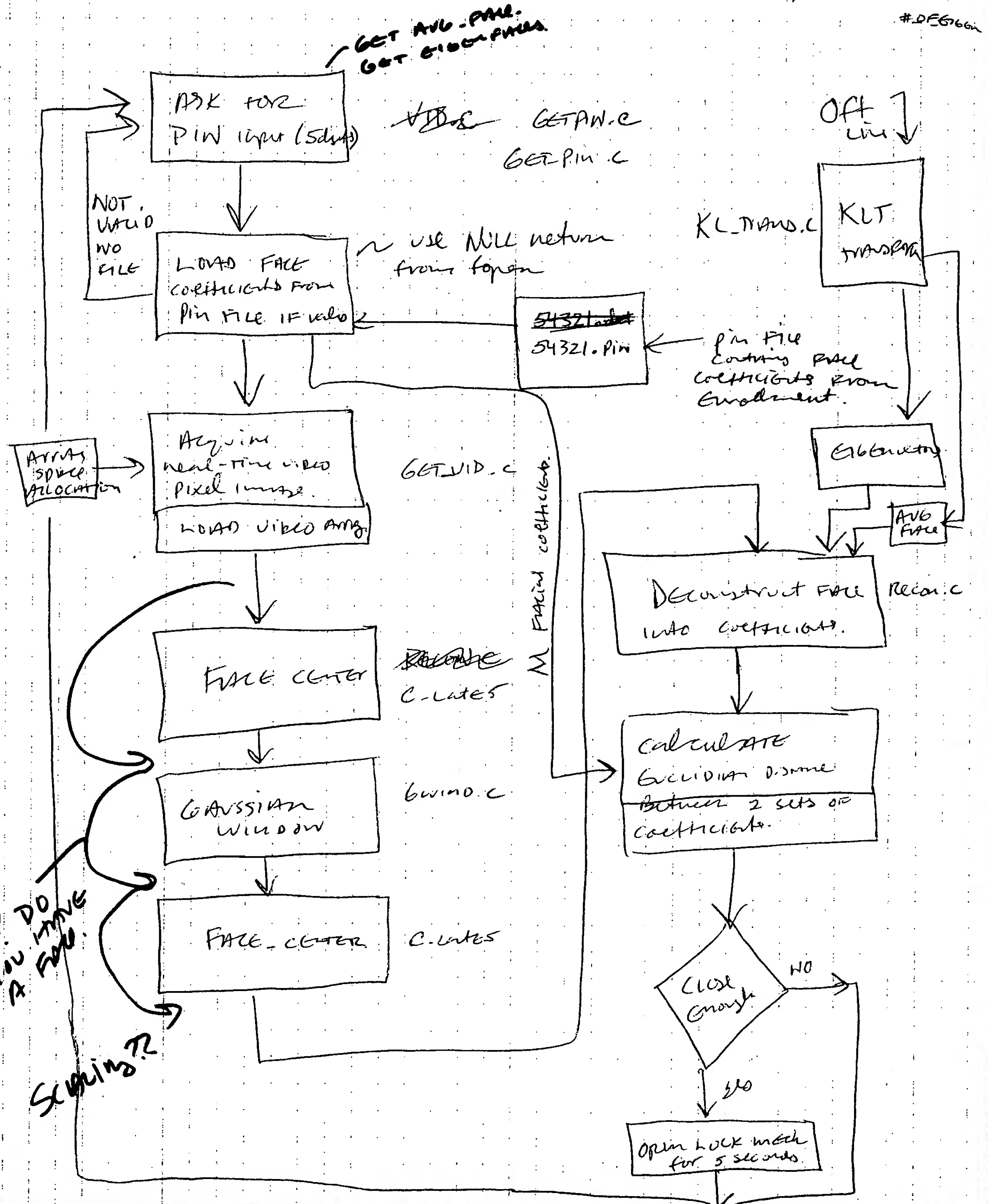
f_ENROLL C (image size, #OF_ EIGENFACES)

SECURE ACCESS SYSTEM, ENROLLMENT



SECURE AREA ACCESS SYSTEM
PROGRAM FLOW-CHART

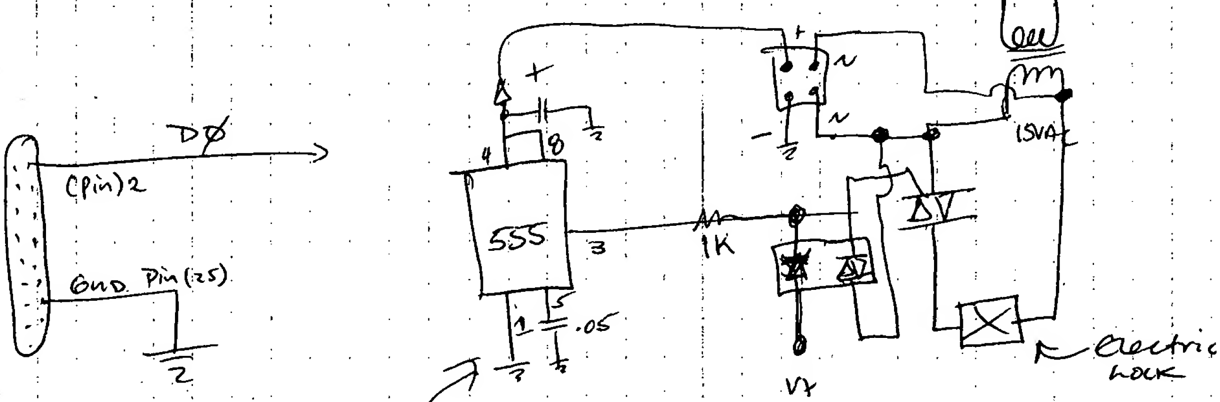
-f verify -Threshold



Secure Access System continued

1/12/92

Lock mechanism can operate off printer port:

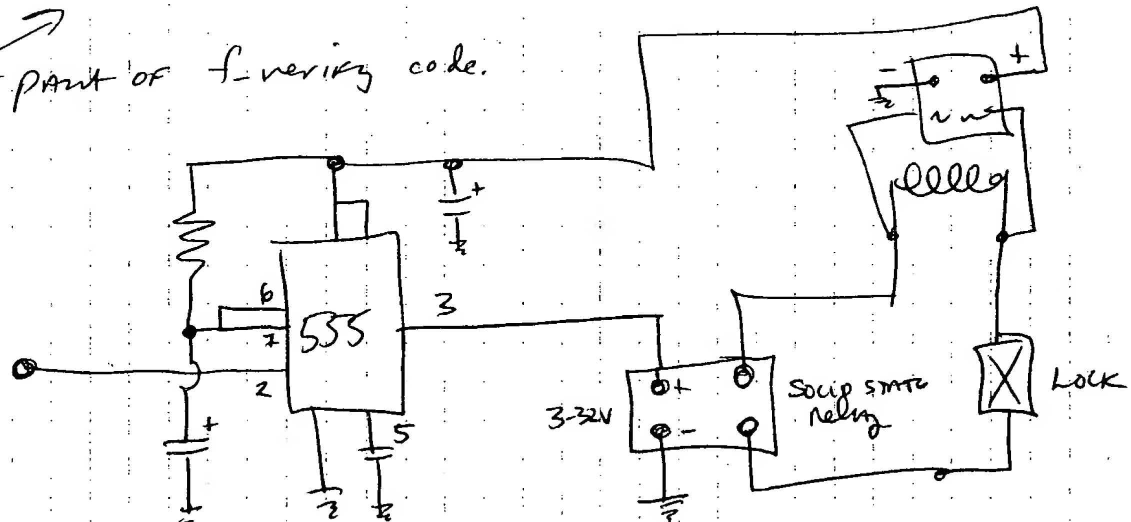


Configure as a one-shot w/ 5 second delay.

Use the outp (0x3F8, 0x__) to control lock — TRIGGER

outp (0x3F8, 0x__) to restore

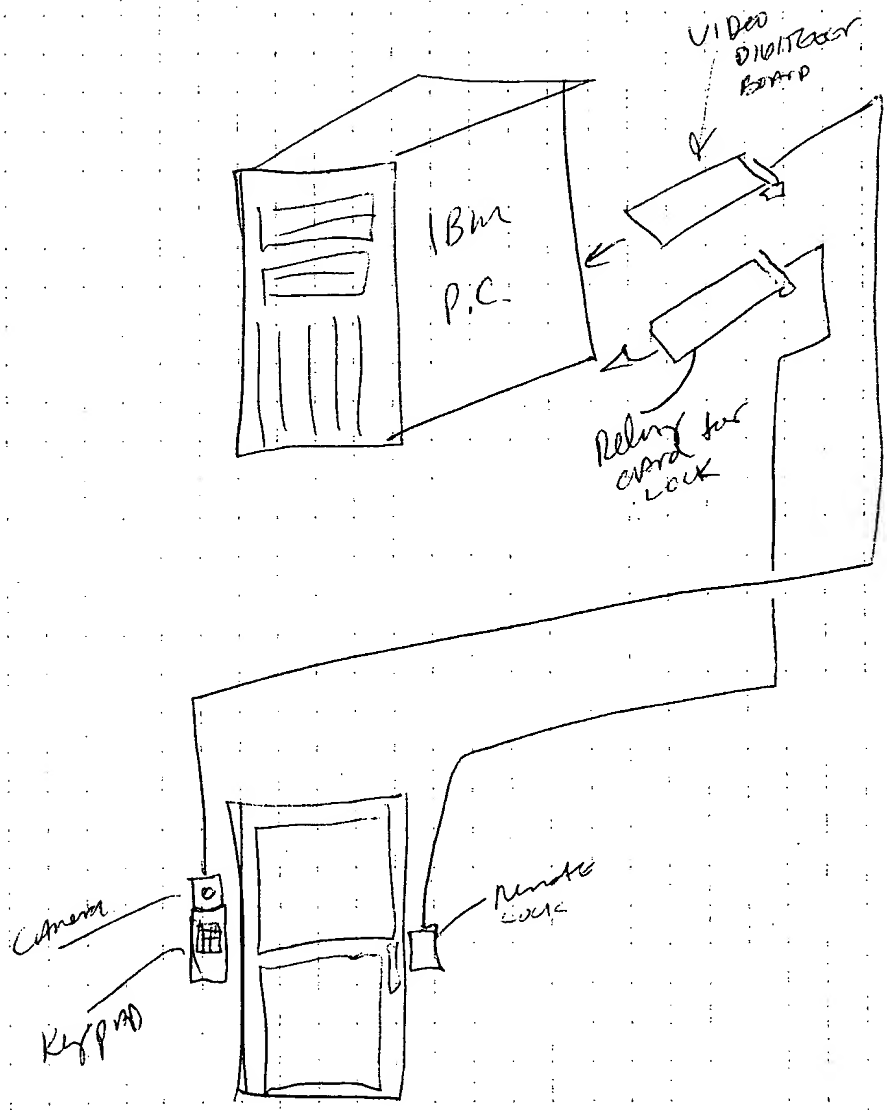
Last part of f_verify code.



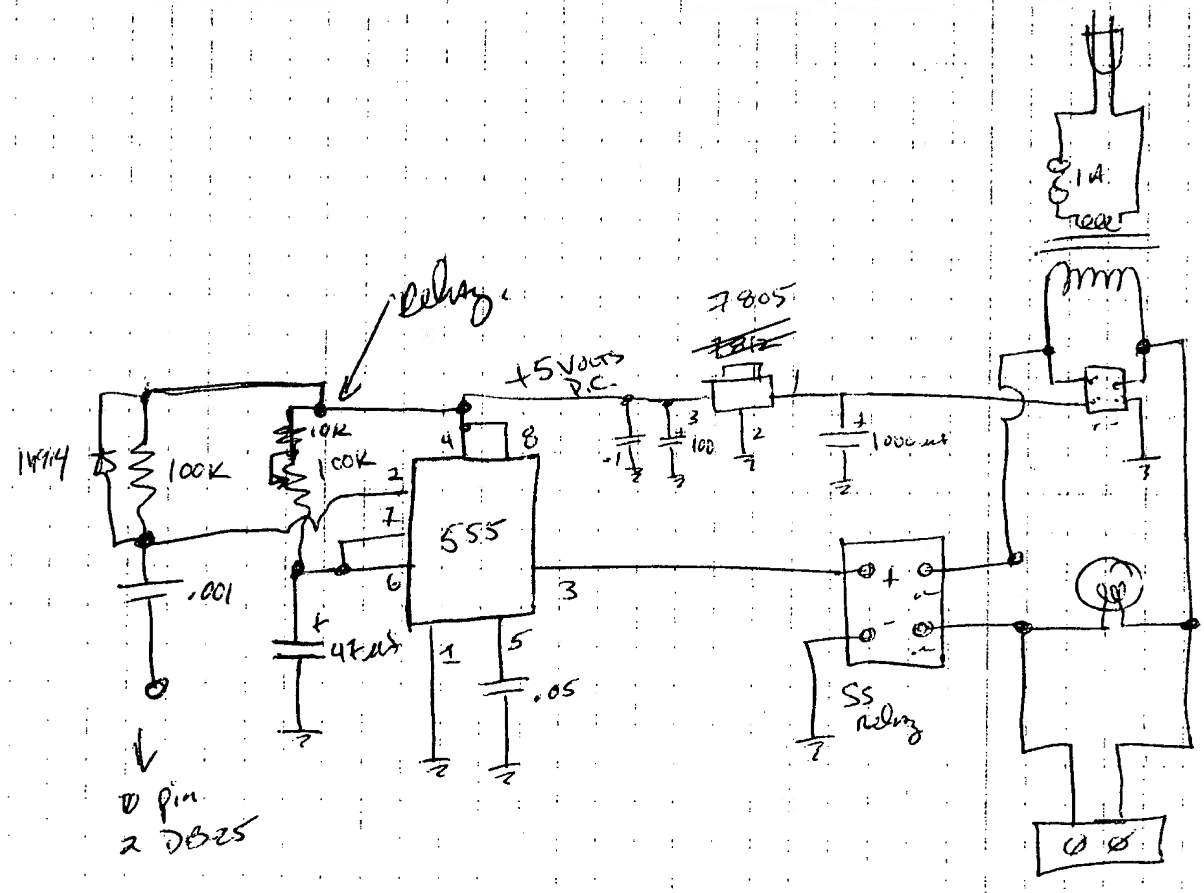
$t \approx R \cdot C$ $= 1.1RC$?

Preferred Design

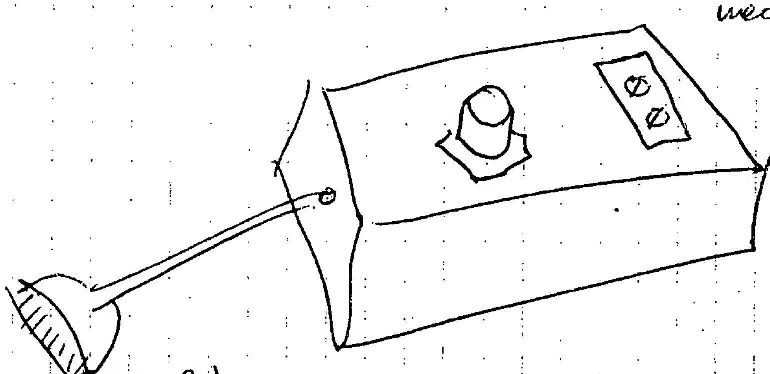
SECURE ACCESS SYSTEM.

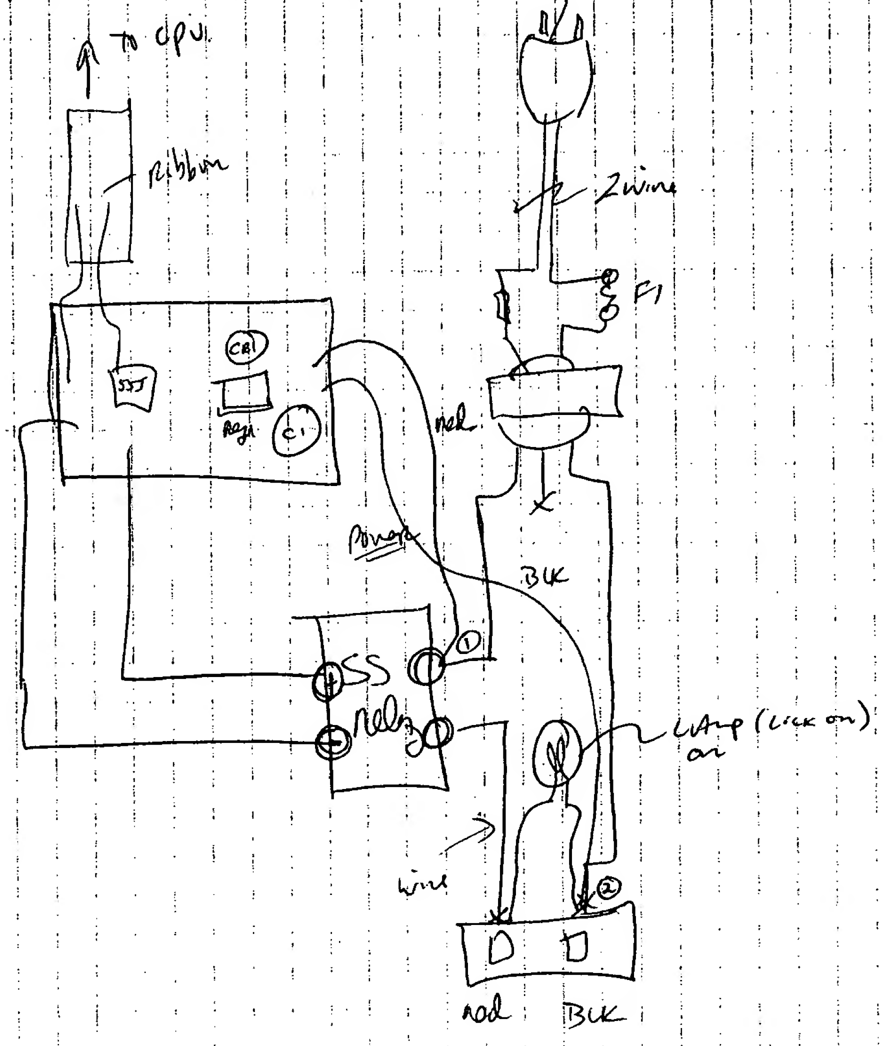


[redacted]



LOCK-INTERFACE control
mechanism





Wiring Diagram Secure access system
Electronic Door release/CPU interface

7

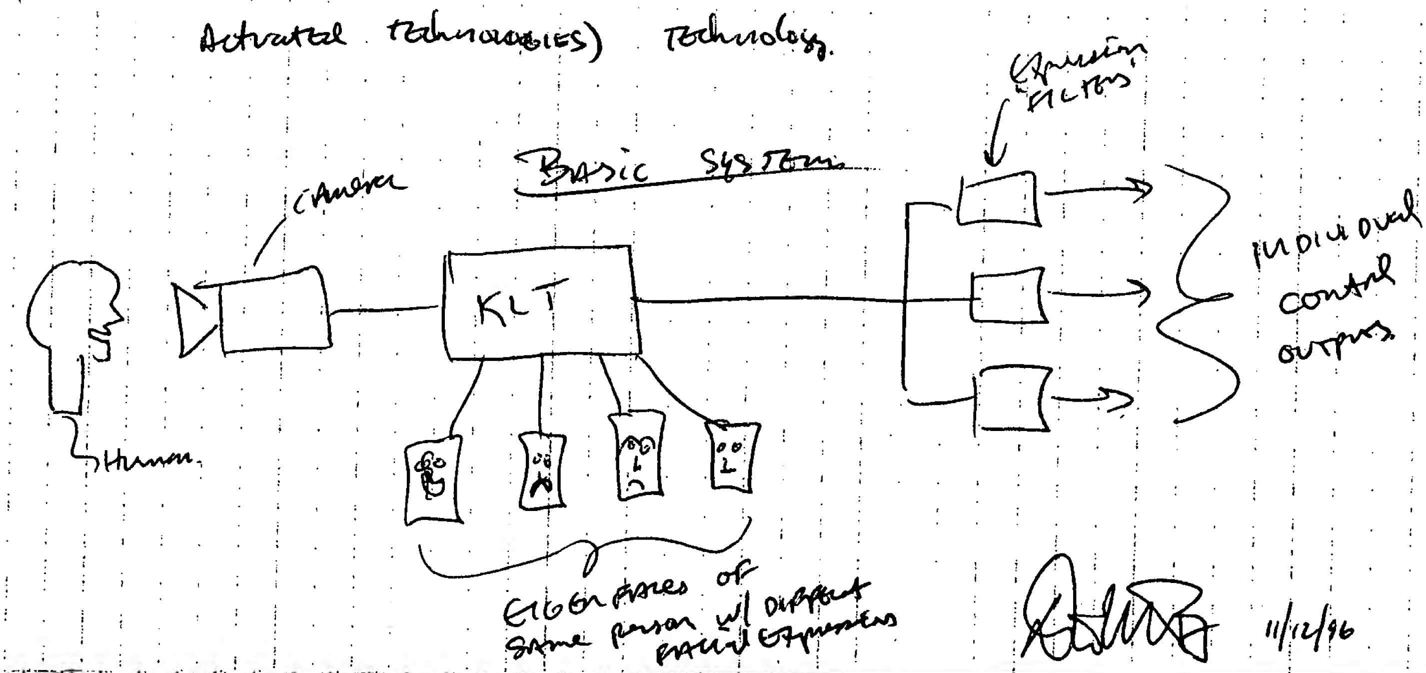
FACE TOY

November 12, 1996

1. Q: Can FaceKey be utilized to detect facial expressions such as smile, frown, mad, happy etc. by using Eigenface approach?

2. Face will be stored in data base w/ varying expressions to see if the Eigen face approach can detect the differences in the covariance matrix such that a set of Eigen faces can be created which represents each instance of the face w/ given facial expression.

3. Face/Feature recognition (see notebook entry 21 Nov 89) could be utilized for facilitated communication + also be utilized for toy applications such as for video game control in conjunction with BAT (Brain Activated Technologies) Technology.

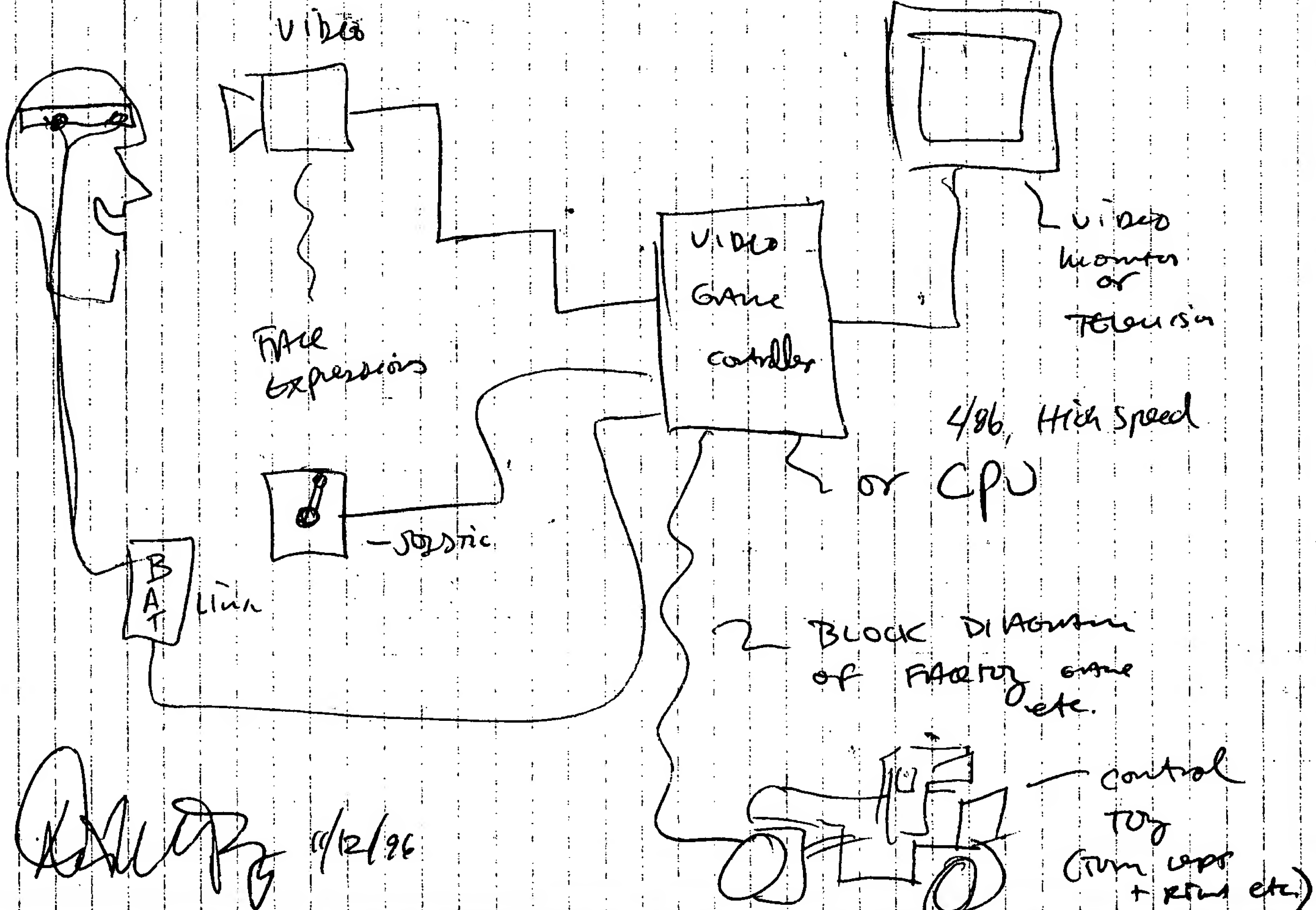


11/12/96

FACETOY / FACILITATED communication Continued.

11/12/96

EACH OUTPUT "FILTER" will TRIGGER A specific event when the FACE is detected. For facilitated communication, the "smile" FACE could TURN on A LAMP, the "Frown" FACE could TURN on A FAN etc. For A VIDEO GAME or TOY, the "smile" FACE could control some activity OF the TOY, while the "Frown" could control another. THIS control could be combined in A VIDEO GAME with A (BAT) LINK interface. Also, the TOY/VIDEO GAME could be connected to the Internet + interacted w/ sending + receiving Emails etc. Also use to search websites. [I have been "playing" with the Internet since this summer + Am only JUST beginning of its total potential].



11/12/96

IVS Face Recognition Toy Software Development
Work Product and Chronology
2/14/05

The CD attached to this document contains the work product (software programs, files & data) of the Face Recognition Toy from 1996 through 1999. The files are stored in folders by year: IVS1996, IVS1997, IVS1998, IVS1999. Each folder contains files generated in their respective calendar-year. Each file has its original date of last modification. To view the dates, right-click on any file and select "properties". The Creation Date is the day the file was copied to the CD and thus 'created'. The Modified Date is the day the program was last modified and is the same as the original 'creation' date for the file. The Accessed Date is today's date, the date the file was last viewed or opened.

Folder: IVS1996

File Name	Date of Creation	Function
IVSFace.exe	7/15/96	Software for finding, aligning, normalizing. Also able to match a face with one in a database. (No real-time capability). *Note: This work predates the development of the Toy algorithms but was the foundation upon which other algorithms were developed.*
FR32.exe	9/4/96	Uses feature recognition (i.e., expressions) to locate Eyes and align face with an optimal matching template. First in the series of algorithms developed specifically for Toy application.
FR32a.exe	9/14/96	Updated version of FR32.exe
FR32b.exe	11/1/96	Updated version of FR32a.exe (Semi-automatic not quite real-time capability).
IVS.dbs	8/17/96	Database of Eigenface coefficients used in early S/W development.
_isreg32.dll	4/29/96	Dynamic Link Library (DLL) used by above executables.
Cvidcap.dll	12/28/96	Dynamic Link Library (DLL) used for capturing and tracking a face in a video image.
Reader32.dll	3/5/96	Dynamic Link Library (DLL) used by above executables.
Images (Folder)	6/10/96	Original spanning set of facial images utilized in training Neural Network algorithms and creating Eigenfaces & Eigenfeatures.

Folder: IVS1997

File Name	Date of Creation	Function
FR32c.exe	1/9/97	Updated version of FR32b.exe.
FR32d.exe	6/21/97	Near Real-Time automatic face finding & matching. Updated version of FR32c.exe
FR32e.exe	7/27/97	Updated version of FR32d.exe
VideoCap.exe	5/15/97	Software for locating and capturing a face in a video image. See example file Vcap.bmp below.
Data0 – Data9	1/8/97	Database of Eigenface coefficients used in early S/W development.
Disk1 – Disk3.zip	6/29/97	Installation package for FR32x.exe – used to install the software for use in prototype development. Files zipped created 6/24/97.
Express.dbs	1/8/97	Facial expression database – early work to create Eigenfeatures to recognize smile, frown, etc.
Tone.wav	10/14/97	Tone.wav is an early file used in testing and developing the use of audio 'control' of an external device (Toy). A DTMF tone file.
Vcap.bmp	5/27/97	A facial image captured and stored using VideoCap.exe. Notice a great photo of the still "young" inventor.

Folder: IVS1998

File Name	Date of Creation	Function
1.ICO – 11.ICO	11/18/98	Icon files of Teddy Bears, used in developing the software for the Toy. Icons were used with program features such as buttons, screens & graphics.
Bubba.wav	10/22/98	Animation Script & Control .wav file. Notice one track of audio and one track of tone-control signals. This file was used when a toy animal was recognized.
Dolly.wav	10/22/98	Animation Script & Control .wav file. Notice one track of audio and one track of tone-control signals. This file was used when a toy doll was recognized.
Happy.wav	10/22/98	Animation Script & Control .wav

		file. Notice one track of audio and one track of tone-control signals. This file was used when a happy face (smiling subject) was seen by the toy (after recognition)
Sad.wav	10/22/98	Animation Script & Control .wav file. Notice one track of audio and one track of tone-control signals. This file was used when a sad face (frowning subject) was seen by the toy (after recognition).
Woody.wav	10/22/98	Animation Script & Control .wav file. Notice one track of audio and one track of tone-control signals. This file was used when a toy action-figure was recognized.

Folder: IVS1999

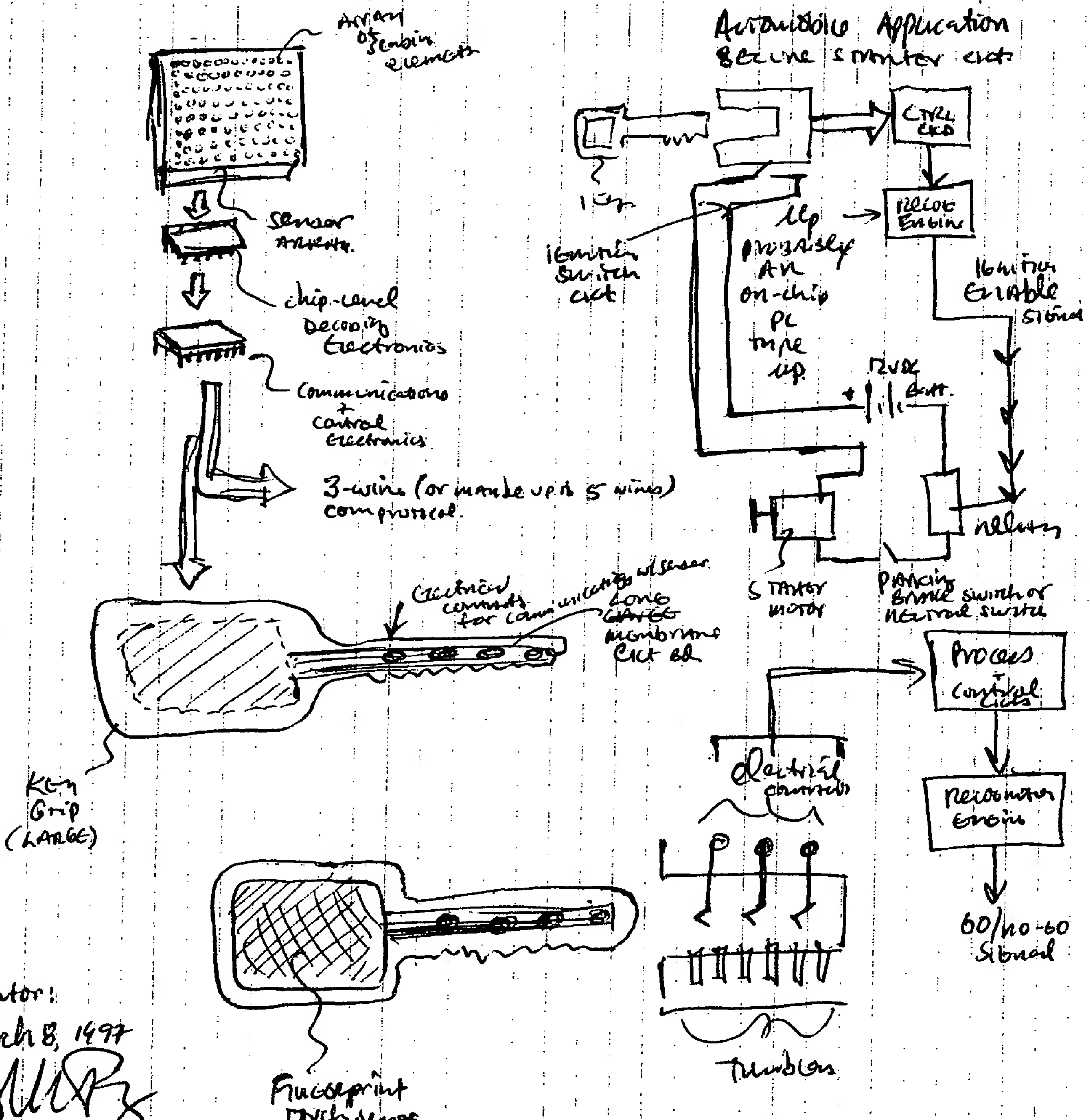
File Name	Date of Creation	Function
FaceKey2.exe	2/23/99	Early compilation of the final configuration for the Toy Face Recognition Software. Almost identical to the software used in the current Toy demonstration (as seen in the video).
Bubba.pk	7/30/99	PK amplitude file generated by the audio workstation to automatically adjust volume levels.
Dad.wav	7/28/99	.wav file played when the toy recognized the inventor (used in demonstrations and can be heard in the video).
Herewego.wav	7/28/99	An interactive .wav file used while Jacob was surfing the internet with the Toy acting as a 'Web Nanny'.
Jacob.wav/Jacob.pk	7/28/99	.wav file played when the toy recognized the inventor's son Jacob. *Note: The audio refers to Jacob as being "3 years old". Jacob was born in 1995 corroborating this "Jacob" wave file was created prior to Sept 1999.*
Jacob1.wav/Jacob1.pk	7/26/99	An interactive .wav file used while Jacob was surfing the internet with the Toy acting as a 'Web Nanny'.

Jonathan.wav/Jonathan.pk	7/28/99	.wav file played when the toy recognized the inventor's son Jonathan. *Note: The audio refers to Jonathan as being "6 years old". Jonathan was born in 1992 corroborating this "Jonathan" wave file was created prior to Nov 1999.*
Jonathan1.wav/Jonathan1.pk	7/28/99	An interactive .wav file used while Jonathan was surfing the internet with the Toy acting as a 'Web Nanny'.
Squeeze.wav/Squeeze.pk	7/28/99	An interactive .wav file used to control the actions of the internet after the subject had been recognized.

3-8-97 Dwt. Capacitive Fingerprint w/ Key (See early concept Dec 15, 88)

Earlier concept for key system utilized fibre optics w/ backlight ~ not practical for manuf. New idea based on a non-optical sensor means utilizing a touch sensor (electrostatic or capacitive)

[I read about early sensors in a trade mag I think it was EE Times]



Inventor:
March 8, 1997

Witness:

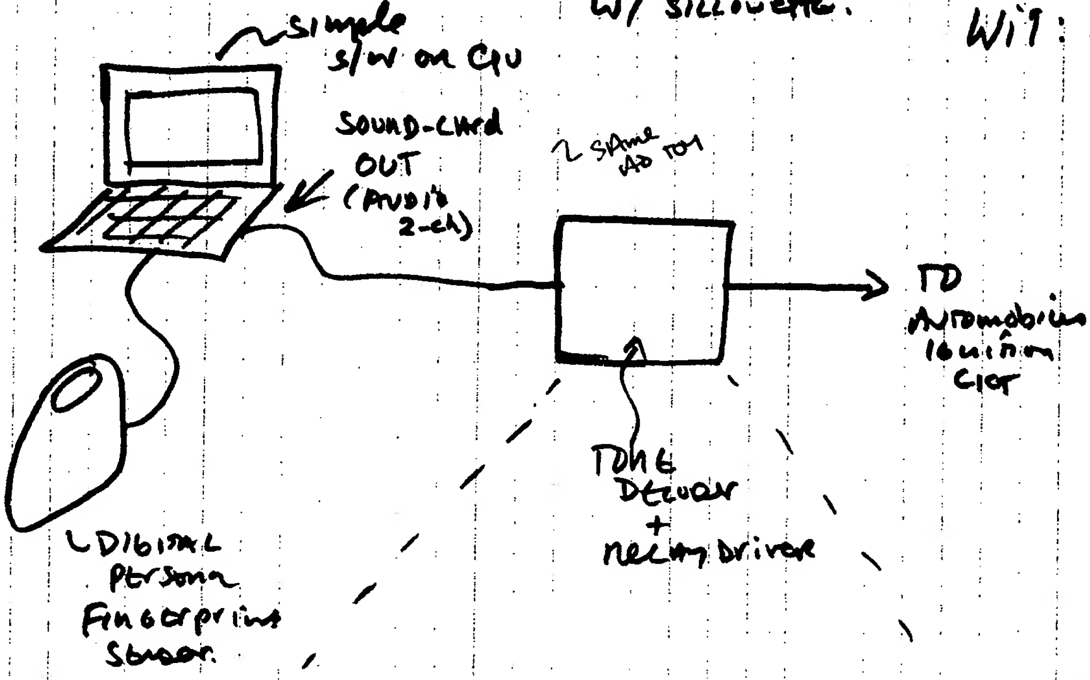
9/11/97

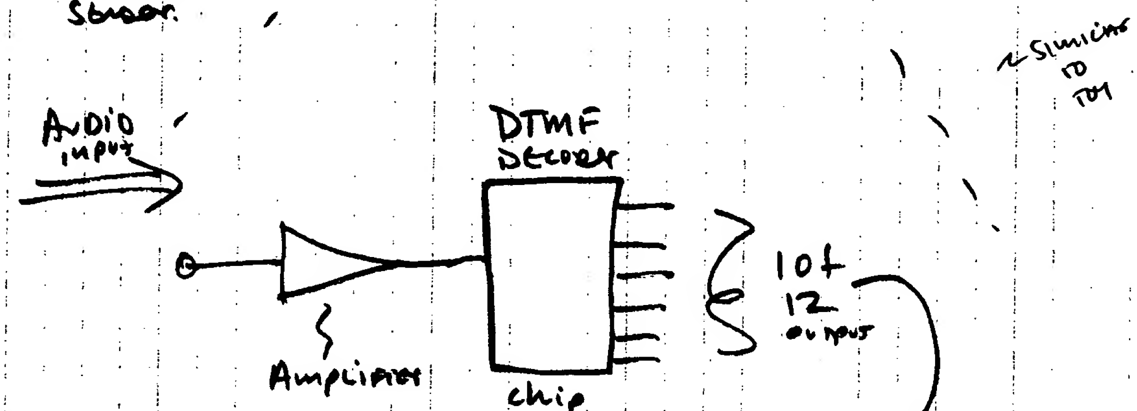
Fingerprint/Key System

9/11/97

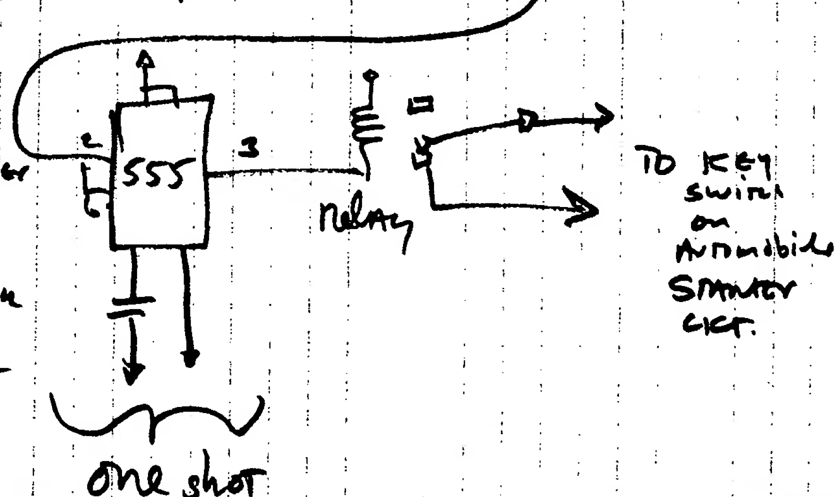
Wit:

TEST System
w/ silhouette.





NOTE:
Software Generates
A DTMF tone
when fingerprint
is recognized triggering
555 monostable for
3 seconds which in-turn
engages car's ignition
switch + system to start
car.

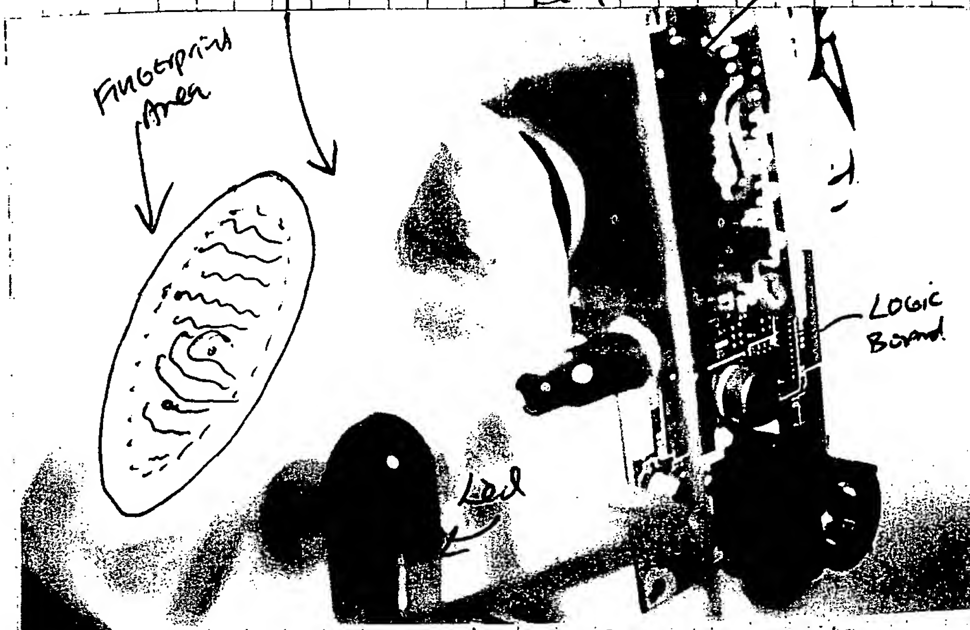


FP-KEY PROJECT

Lens Optic

Photo of prototype Key

CCD Camera



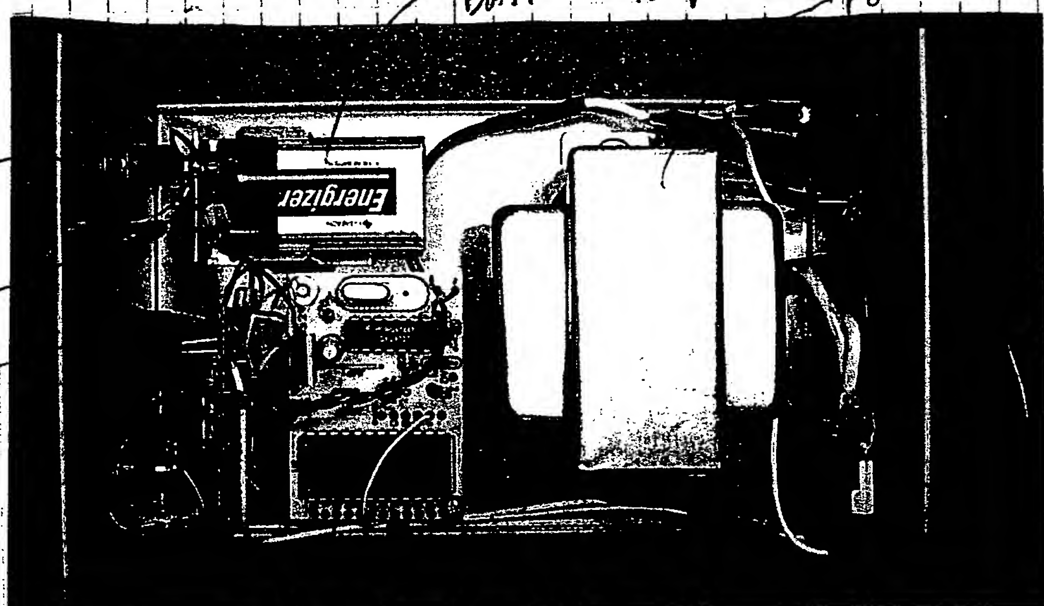
BATTERY (For portable use)

Power transformer

On/off Switch

Power Led

Ignition Relay



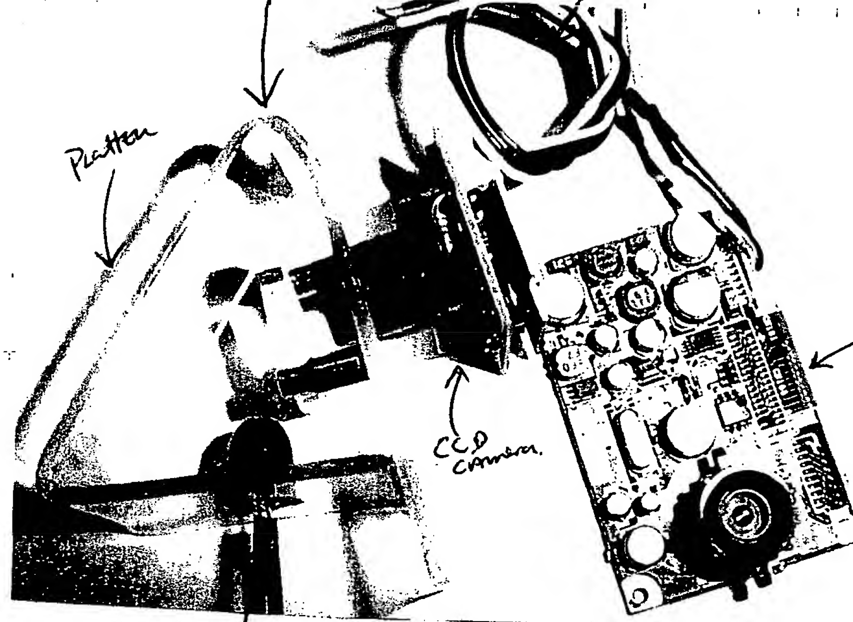
Ignition Switch

Photo of prototype Box

DTMF Decoder

As used w/ 97 CCD Silhouette VAN

1/5/98 [signature] Witness: [signature]



Optical Housing

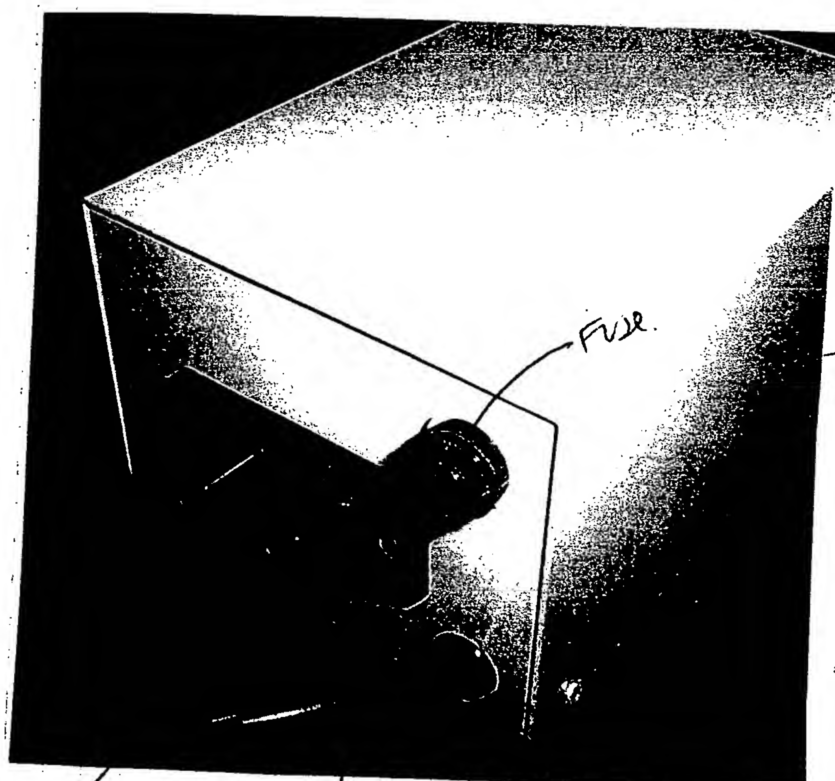
Wires on key connector.

Platten

Logic + Decoder ckts.

CCD camera.

Leds



Fuse.

Box to mount under dash.

Connector for Ignition switch

Charger cord

12

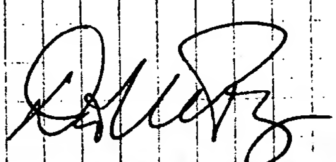
3/21/1998 for F.P. KG software (written in C++)

```
 _wstr = new WCHAR[len];

 //              CP_ACP
 MultiByteToWideChar (CP_OEMCP, 0, s, -1, _wstr, len);
#endif
}

AtoWConverter::~AtoWConverter () {
 delete [] _wstr;
}

AtoWConverter::operator LPCWSTR () {
 return _wstr;
}

WtoAConverter::WtoAConverter (LPCWSTR w) {
#ifdef UNICODE
 // In this case both strings are Unicode and we just use normal lstrcpy.
 TCHAR* _astr = new TCHAR[lstrlen(s)];
 return lstrcpy (_astr, s);
#else
 // We need to convert the Unicode to ASCII string and copy after that.
 //                CP_ACP
 int len = WideCharToMultiByte (CP_OEMCP, 0, w, -1, NULL, 0, NULL, NULL);
 _astr = new char[len];

 //              CP_ACP
 WideCharToMultiByte (CP_OEMCP, 0, w, -1, _astr, len, NULL, NULL);
#endif
}

WtoAConverter::~WtoAConverter () {
 delete [] _astr;
}

WtoAConverter::operator LPCTSTR () {
 return _astr;
}

//Global variables

int g_nWidth, g_nHeight;            //width and height of drawing area
FT_HANDLE g_FtContext;              //dpFpFns context
FT_HANDLE g_dbContext;              //dpDbase context
DB_USERID g_currentUserID;          //current user ID
FT_UI_LINK g_FtUILink;              //UI link information
FT_DEVICE_INFO g_DevInfo;           //device info
FT_BYTE* g_Features;                //features
int g_recommendedFtrLen, g_minFtrLen;  //recommended features length and minimu
int g_nFingerCount;                 //variable to count down fingers left while registering
BOOL g_bIsVerifying;                //verifying or register mode, in verifying mode do
CRect g_rectDrawArea;               //rectangle to draw fingerprints in dialog
CWinThread* g_pRegisterThread = NULL;  //pointer to register thread
CWinThread* g_pVerifyThread = NULL;    //pointer to verify thread
BYTE *g_FPBuffer;                   // Pointer to buffer which holds data portion of bitn
                                    // This pointer is provided for toolkit in the callback function.
                                    // Toolkit loads image in the buffer and notifies GUI thread th
HWND g_hwnd = NULL;                 //HWND of dialog
BOOL g_bIsTimeToDie = FALSE;        //variable used to ensure clean exit from pr
CEvent g_event(FALSE, TRUE);        //event used to make sure that program will r
                                    //while toolkit is doing something (like waiting for image)
                                    //second argument TRUE for manual CEvent
```

```
public:
  CAboutDlg();

// Dialog Data
  //{{AFX_DATA(CAboutDlg)
  enum { IDD = IDD_ABOUTBOX };
  //}}AFX_DATA

  // ClassWizard generated virtual function overrides
  //{{AFX_VIRTUAL(CAboutDlg)
  protected:
  virtual void DoDataExchange(CDataExchange* pDX);   // DDX/DDV supp
  //}}AFX_VIRTUAL

// Implementation
protected:
  //{{AFX_MSG(CAboutDlg)
  //}}AFX_MSG
  DECLARE_MESSAGE_MAP()
};

CAboutDlg::CAboutDlg() : CDialog(CAboutDlg::IDD)
{
  //{{AFX_DATA_INIT(CAboutDlg)
  //}}AFX_DATA_INIT
}

void CAboutDlg::DoDataExchange(CDataExchange* pDX)
{
  CDialog::DoDataExchange(pDX);
  //{{AFX_DATA_MAP(CAboutDlg)
  //}}AFX_DATA_MAP
}

BEGIN_MESSAGE_MAP(CAboutDlg, CDialog)
  //{{AFX_MSG_MAP(CAboutDlg)
    // No message handlers
  //}}AFX_MSG_MAP
END_MESSAGE_MAP()

/////////////////////////////////////////////////////////////////////////////
// CFTSampleDBDlg dialog

CFTSampleDBDlg::CFTSampleDBDlg(CWnd* pParent /*=NULL*/)
  : CDialog(CFTSampleDBDlg::IDD, pParent)
{
  //{{AFX_DATA_INIT(CFTSampleDBDlg)
  m_strPrompt = _T("");
  //}}AFX_DATA_INIT
  // Note that LoadIcon does not require a subsequent DestroyIcon in Win32
  m_hIcon = AfxGetApp()->LoadIcon(IDR_MAINFRAME);

}

void CFTSampleDBDlg::DoDataExchange(CDataExchange* pDX)
{
  CDialog::DoDataExchange(pDX);
  //{{AFX_DATA_MAP(CFTSampleDBDlg)
  DDX_Text(pDX, IDC_EDIT_PROMPT, m_strPrompt);
  //}}AFX_DATA_MAP
}

BEGIN_MESSAGE_MAP(CFTSampleDBDlg, CDialog)
  //{{AFX_MSG_MAP(CFTSampleDBDlg)
  ON_WM_SYSCOMMAND()
  ON_WM_PAINT()
  ON_WM_QUERYDRAGICON()
  ON_BN_CLICKED(IDC_RADIO_REGISTER, OnRadioRegister)
  ON_BN_CLICKED(IDC_RADIO_VERIFY, OnRadioVerify)
  ON_WM_DESTROY()
```

```
}
n
}

FT_
{
 T
 re
}

FT_
{
 T
 re
}

//Th
UIN
{
 g_
 H
 hv
 ::S
 F
 F
 g_

 rc

 g_
 ::I
```

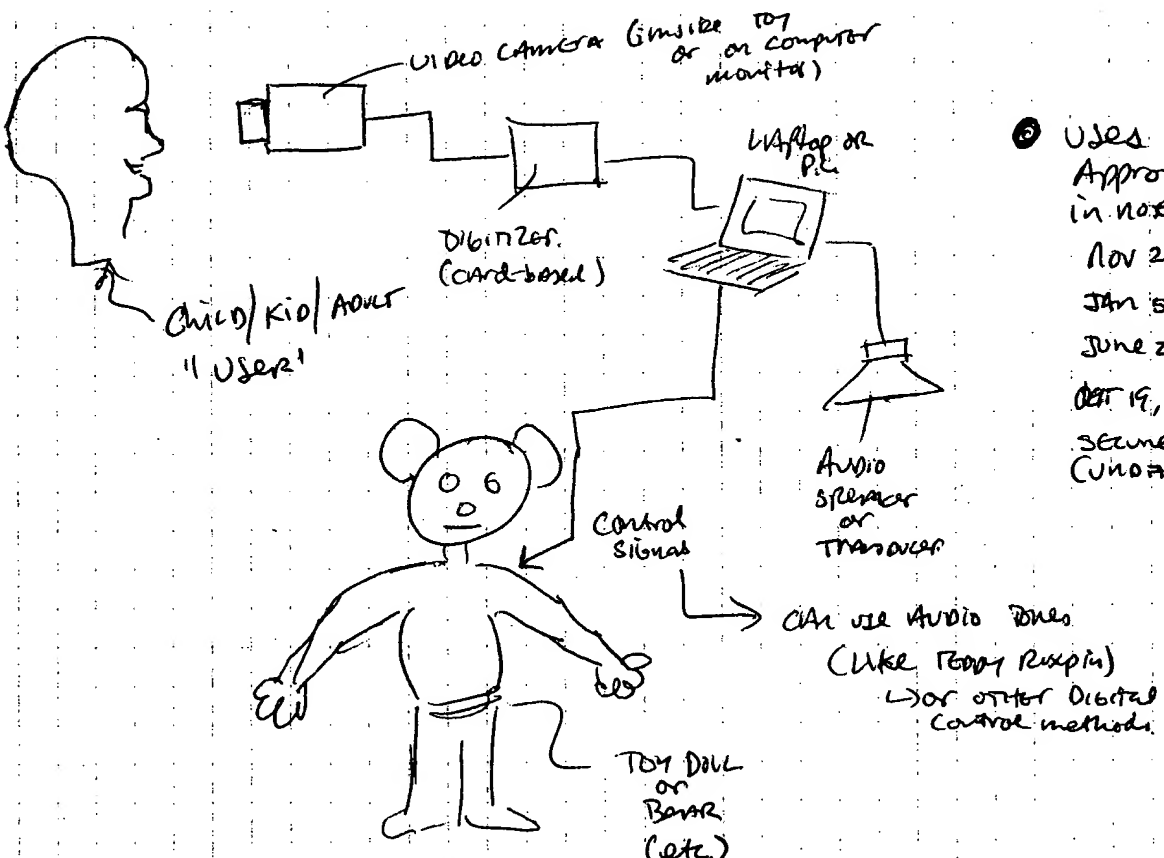
```
    switch(pStatus->code)
    {
      case FT_READY_TO_FILL_BUF:
        SendMessage(g_hwnd, MESSAGE_READY_TO_FILL_BU
        TRACE("FT_READY_TO_FILL_BUF\n");
        break;

      case FT_BUF_FILLED:
        SendMessage(g_hwnd, MESSAGE_IMAGE_RECEIVED, 0
        TRACE("FT_BUF_FILLED\n");
        break;

      case FT_IMAGE_INFO:       // param1 = FT_IMGQUALITY
        SendMessage(g_hwnd, MESSAGE_FT_IMAGE_INFO, 0, (
        TRACE("FT_IMAGE_INFO\n");
        break;

      case FT_FEATURES_INFO:
        SendMessage(g_hwnd, MESSAGE_FT_FEATURES_INFO,
        TRACE("FT_FEATURES_INFO\n");
        break;

      case FT_WAITING_FOR_IMAGE:
        SendMessage(g_hwnd, MESSAGE_FT_WAITING_FOR_IM
        TRACE("FT_WAITING_FOR_IMAGE\n");
        break;

      default:
        break;
    }
    return FT_ID_CONTINUE;
  }

FT_DISPLAY_BUF_PT getRegisterDisplayBuf (FT_FTR_TYPE, vo
{
  TRACE("getRegisterDisplayBuf\n");
  return g_FPBuffer;
}

FT_RETCODE releaseRegisterDisplayBuf (FT_DISPLAY_BUF_PT
{
  TRACE("releaseRegisterDisplayBuf\n");
  return 0;
}


//Thread functions
UINT DoRegister(LPVOID pParam)       //register thread function
{
  g_nFingerCount = 0;
  HWND hwnd;
  hwnd = ::GetDlgItem((HWND)pParam, IDC_EDIT_PROMPT);
  ::SetWindowText(hwnd, "4 fingers left");
  FT_RETCODE rc;
  FT_RESULT result;
  g_event.ResetEvent();        //we reset CEvent in nonsignaled st
                               //program will wait until FT_register comple
  rc = FT_register(g_FtContext,
        FT_TRUE,              //allowLearning has to be FT_TRUE
                              //use features with database
        g_recommendedFtrLen,
        g_Features,
        NULL,
        &result);
  g_event.SetEvent();          //now we can exit
  ::PostMessage((HWND)pParam, WM_REGISTERTHREADFINISHED, 0, (LPARAM)result);
  return 0;
}

UINT DoVerify(LPVOID pParam)        //verify thread function
{
  HWND hwnd;
  DB_USERID userID;
  DB_FINGER_KEY fingerKey;
  FT_RETCODE rc;
  FT_VER_SCORE score;
  FT_BOOL bIsVerified;
  FT_RESULT result;
  WCHAR     name[DB_USERNAME_LEN+1];
  FT_IMG_QUALITY qualityImg;
  FT_FTR_QUALITY qualityFtr;
  while(!g_bIsTimeToDie)
  {
    hwnd = ::GetDlgItem((HWND)pParam, IDC_EDIT_PROMPT);
    ::SetWindowText(hwnd, "Please put your finger on sensor");
    DB_topUser(g_dbContext, &userID);
    g_event.ResetEvent();
    rc = FT_acquireFeatures(g_FtContext,
          FT_VER_FTR,
          g_recommendedFtrLen,
          g_Features,
          &qualityImg,
          &qualityFtr,
          &result);
    if(result == FT_SUCCESS)
    {
      g_event.SetEvent();
      hwnd = ::GetDlgItem((HWND)pParam, IDC_EDIT_USERNAME);
      ::SetWindowText(hwnd, "Unknown user");
      hwnd = ::GetDlgItem((HWND)pParam, IDC_EDIT_FINGER);
      ::SetWindowText(hwnd, "");
      while(DB_nextUser(g_dbContext, userID, &userID,
        NULL, NULL, NULL, NULL, NULL) != FT_WRN_EOF)    //loop through the list of users
      {
        g_event.ResetEvent();
        rc = DB_authenticate(g_dbContext, userID, NULL, DB_UNKNOWN, g_Features, NULL,
                    NULL, &fingerKey, &score, NULL, &bIsVerified);
        g_event.SetEvent();
        if(bIsVerified)
        {
          g_event.ResetEvent();
          DB_readUserData(g_dbContext, userID, name, NULL, NULL, NULL, NULL);
          g_event.SetEvent();
          hwnd = ::GetDlgItem((HWND)pParam, IDC_EDIT_USERNAME);
          ::SetWindowText(hwnd, WtoAConverter(name));
          ::PostMessage((HWND)pParam, WM_USERVERIFIED, 0, (LPARAM)fingerKey.finger);
        }
      }
    }
    else
    {
      g_event.SetEvent();
      ::PostMessage((HWND)pParam, WM_VERIFYTHREADFINISHED, 0, 0);
      return 0;
    }

  }
  ::PostMessage((HWND)pParam, WM_VERIFYTHREADFINISHED, 0, 0);
  return 0;
}
```



FACE RECOGNITION w/ TOY (BEAR) or DOLL

12/20/1998 [signature] [signature]

- First concept of toy + face recognition was to use FaceKey S/W to control a small motorized toy like a truck. See entry in note-book dated Nov 12, 1996.
- Original idea was to recognize who the user is + facial expressions.
- What if toy is a bear or doll that connects to a computer (like the truck in the Nov-96 entry) the CPU would control animation of the toy (eyes, mouth, hands etc.) while a computer synthesizer voice or other audio technology interacts w/ the user.
  - Could be used as an interactive toy or game.
  - Could be used as a "web nanny"
  - Could be an entirely new browser technology that allows a kid or young child to experience "surfing" the web safely.
  - The browser is the toy.



- Uses Eigenface approach described in notebook entries
  - Nov 21, 1989
  - Jan 5, 1990
  - June 27, 1991
  - Oct 19, 1991 (locating)
  - Secure access (undated entry)

WIT:

[signature] 12/20/98

FACE RECOGNITION W/ TOY (CONTINUED)

12/20/98

operation (STEPS) [web browser]

1) ENROLL USER, CREATE LIBRARY OF STORED FACES → CONVERT EACH FACE TO A SET OF N-DIMENSIONAL COEFFICIENTS.

- STORE PERSONAL INFO → NAME, AGE, FAVORITE COLOR, etc.

- STORE MULTIPLE IMAGES → HAPPY, SAD, MAD.

↳ USE A "GAME" TO DO THIS: "PLEASE SHOW ME YOUR SILLY FACE... NOW SHOW ME YOUR MAD FACE!"

- STORE IMAGES OF OTHER TOYS THAT HAVE FACES (DOLLS BEARS, ROBOTS etc.)

2) TOY CAN NOW "PLAY" GAMES OR BE UTILIZED AS A web BROWSER. COULD PLAY GAMES LIKE:

- HIDE + GO SEEK
- CHECKERS (PERSONALIZED)
- "Remember" GAMES (memory where the TOY LEARNS STUFF ABOUT YOU)
- INTERACTIVE ADVENTURE GAMES
- INTERACTIVE DIARY ← Remember DAILY ACTIVITIES
- MULTIPLE USER(S) ACROSS INTERNET COULD meet on LINE.

3) TOY WOULD

a) RECOGNIZE USER +/or FACIAL EXPRESSION

b) SELECT AN APPROPRIATE "ANIMATION SCRIPT" FROM A LARGER LIBRARY OF SCRIPTS

c) ANIMATION SCRIPT WOULD INTERACT W/ USER PROVIDING INFINITE POSSIBILITIES + COMBINATIONS

d) TOY WOULD LEARN ABOUT EACH USER + GET SMARTER AS IT GOES ALONG.

e) TOY COULD MONITOR CHILD/USER'S EXPRESSIONS + ADJUST DIFFICULTY OF GAME ACCORDING TO THE NEEDS OF EACH USER

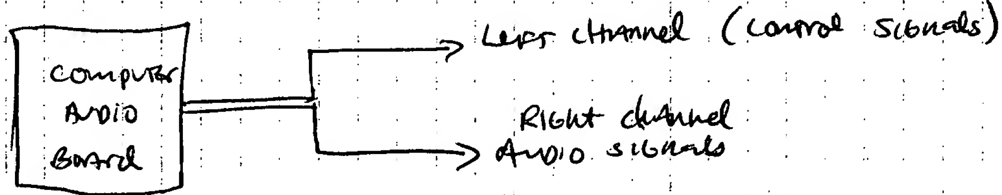
f) TOY COULD ADJUST "BROWSER EXPERIENCE" FOR EACH KID BASED ON THE KNOWN AGE OF EACH USER. (WOULD BE GREAT FOR A BIG FAMILY W/ SEVERAL KIDS + ADULT USERS).

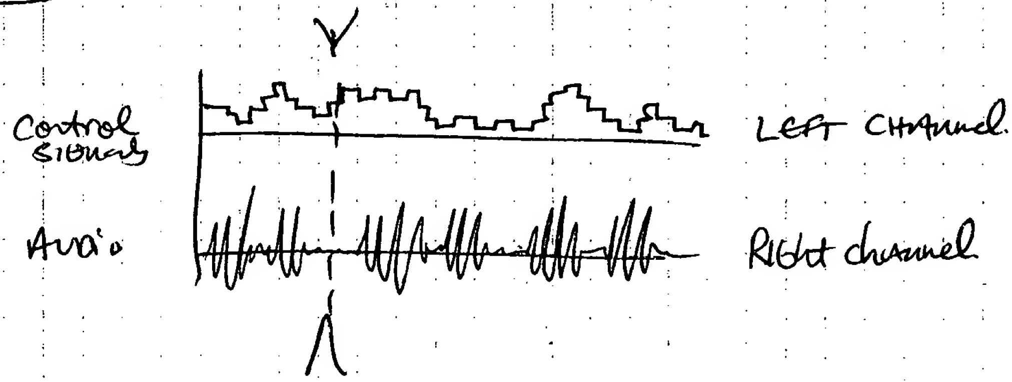
WIT: 12/20/98

FACE RECOGNITION w/ TOY (CONTINUED)

12/20/98

Animation
Control method (with Audio)





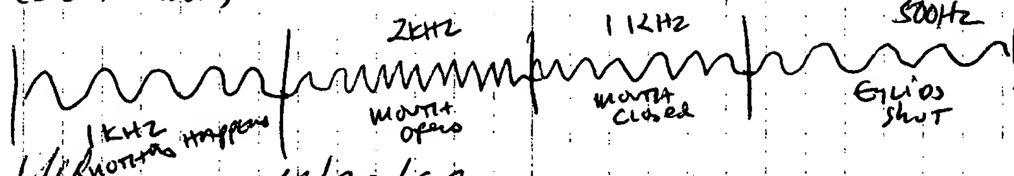
Signals are synchronized so movements coincide w/ Audio track i.e. mouth moves + words come out.

* Use a frequency modulated technique where a starting frequency (F_0) say approx 1 KHz does nothing.

* Frequency deviations create articulation → 1KHz — 2KHz causes the mouth to open: closed at 1khz ~1/2 open @ 1.5khz and fully open @ 2.0khz use Linear control technique

1khz to 0khz (or 500 Hz) causes the eyelids to move: open @ 1KHZ + 1/2 closed @ 900 khz + fully closed @ 500 Hz.

* Thus by varying the "pitch" or frequency of the control signal 2-dimension linear control is obtained.
(Deg of freedom)



WIT: 12/20/98